Internet Technologies: Deliver Web Services Using Sherlock Channels

Session 426

Internet Technologies: Deliver Web Services Using Sherlock Channels

Wayne Loofbourrow
Engineering Manager, Info Apps

Introduction

How to deliver Web Services using Sherlock Channels

- What are Web Services?
- When is a Channel a good choice?
- How to get started
- What's new in Panther?

What You'll Learn

- Overview
  - What's new in Sherlock
  - Developer opportunity
  - Structure of a Sherlock channel
- How to
  - Develop a Channel in four steps
  - What's new for Sherlock developers
  - Advanced topics

What's New In Sherlock

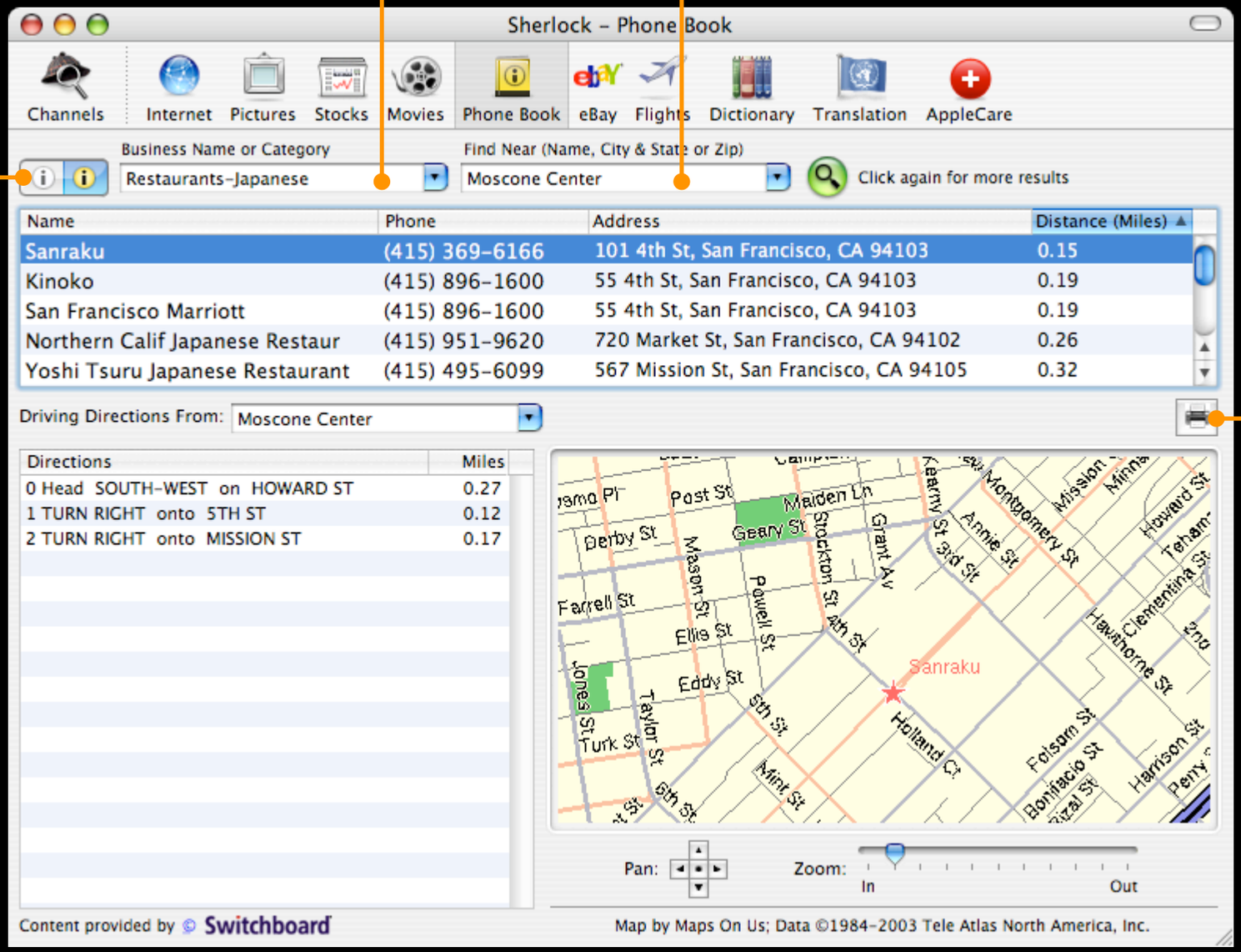
- Enhanced channels
  - Yellow Pages -> phone book
- Performance
  - XQuery 2–3 times faster
  - Launch time
  - Channel switching
- Channel management
  - No more toolbar overload
  - Built-In third party channels

Phone Book Channel

Business Name or Category

Find Near Anyone in Your Address Book

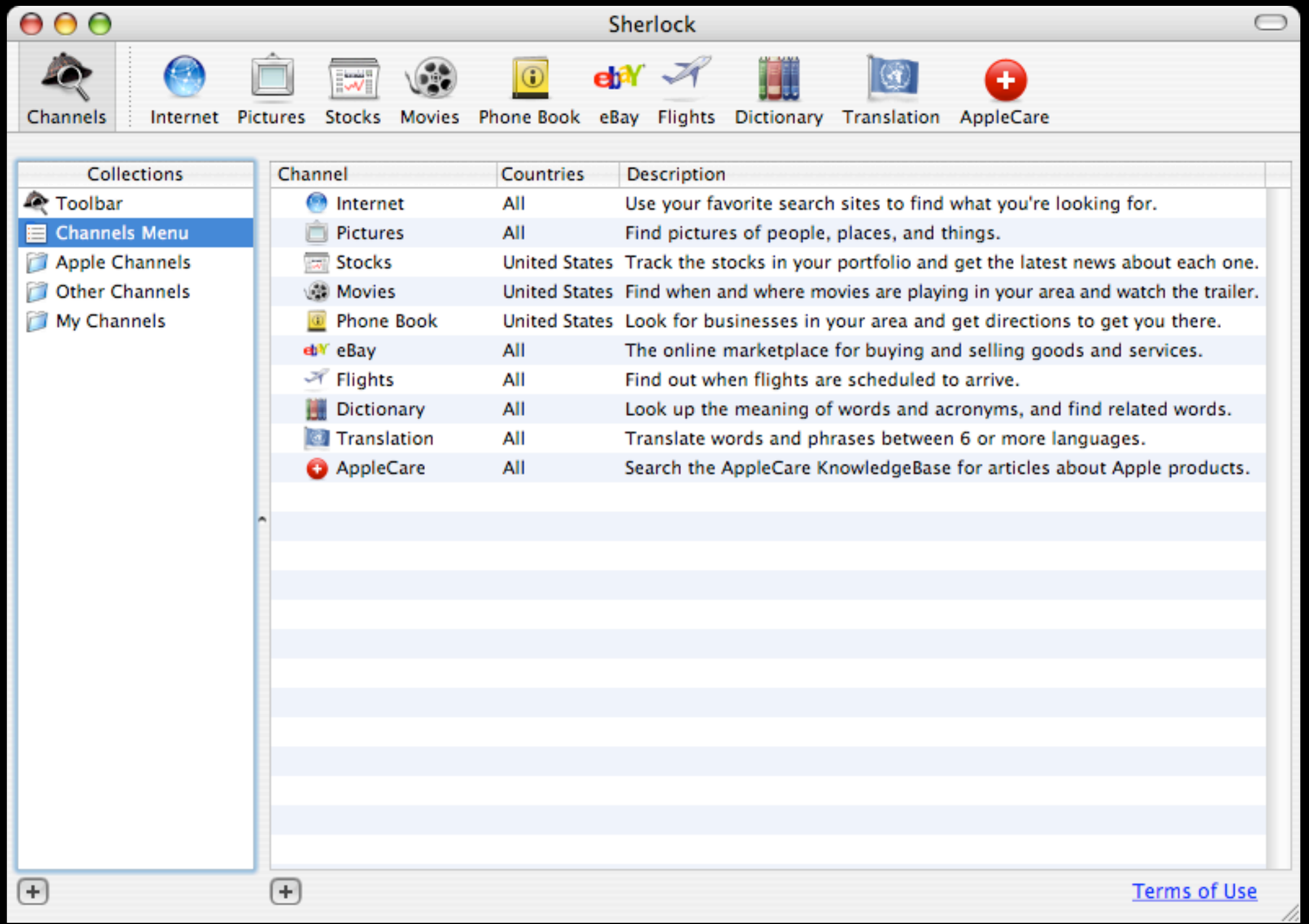
White or Yellow



One Button Print

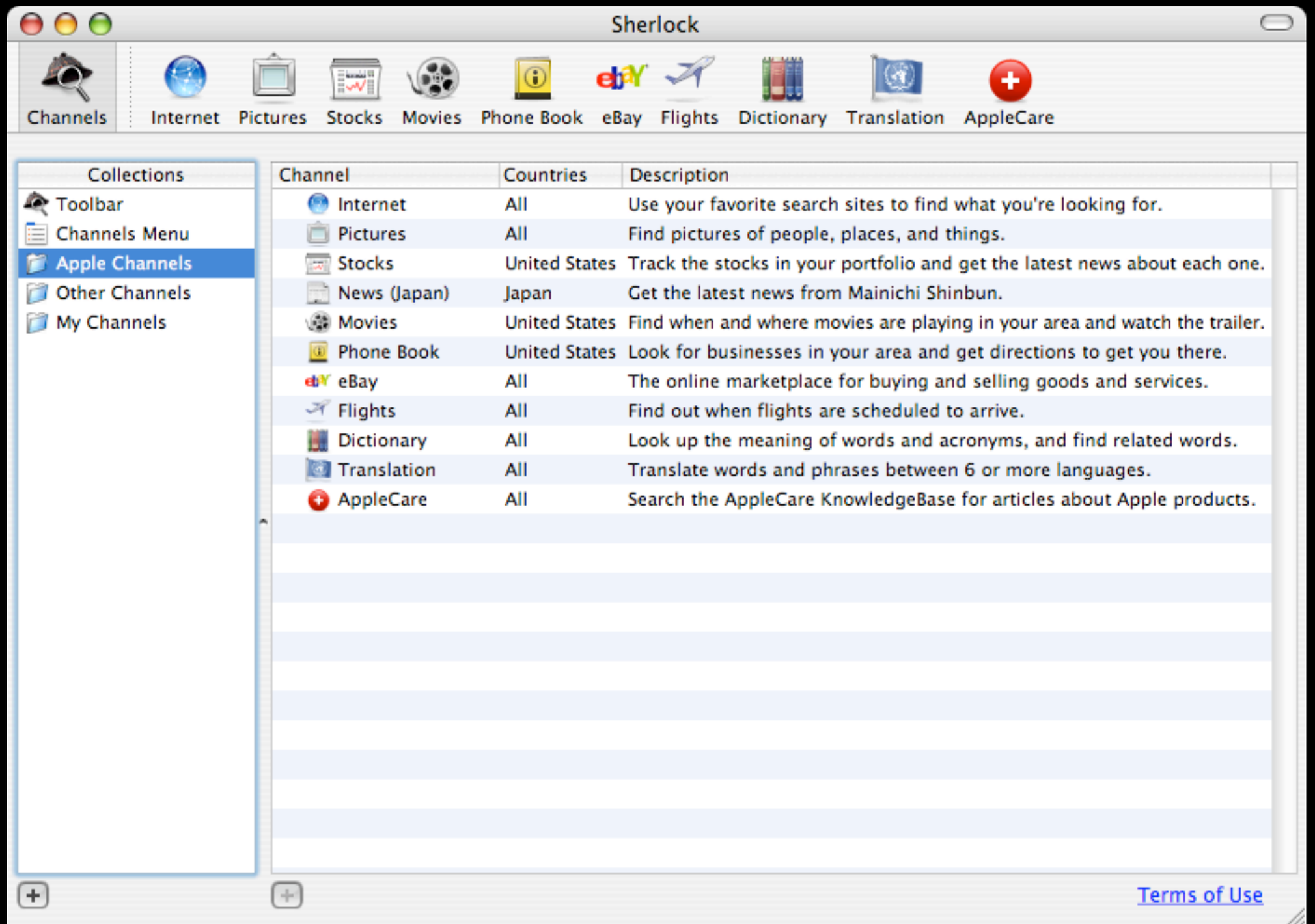
Channels View

Toolbar and Menu



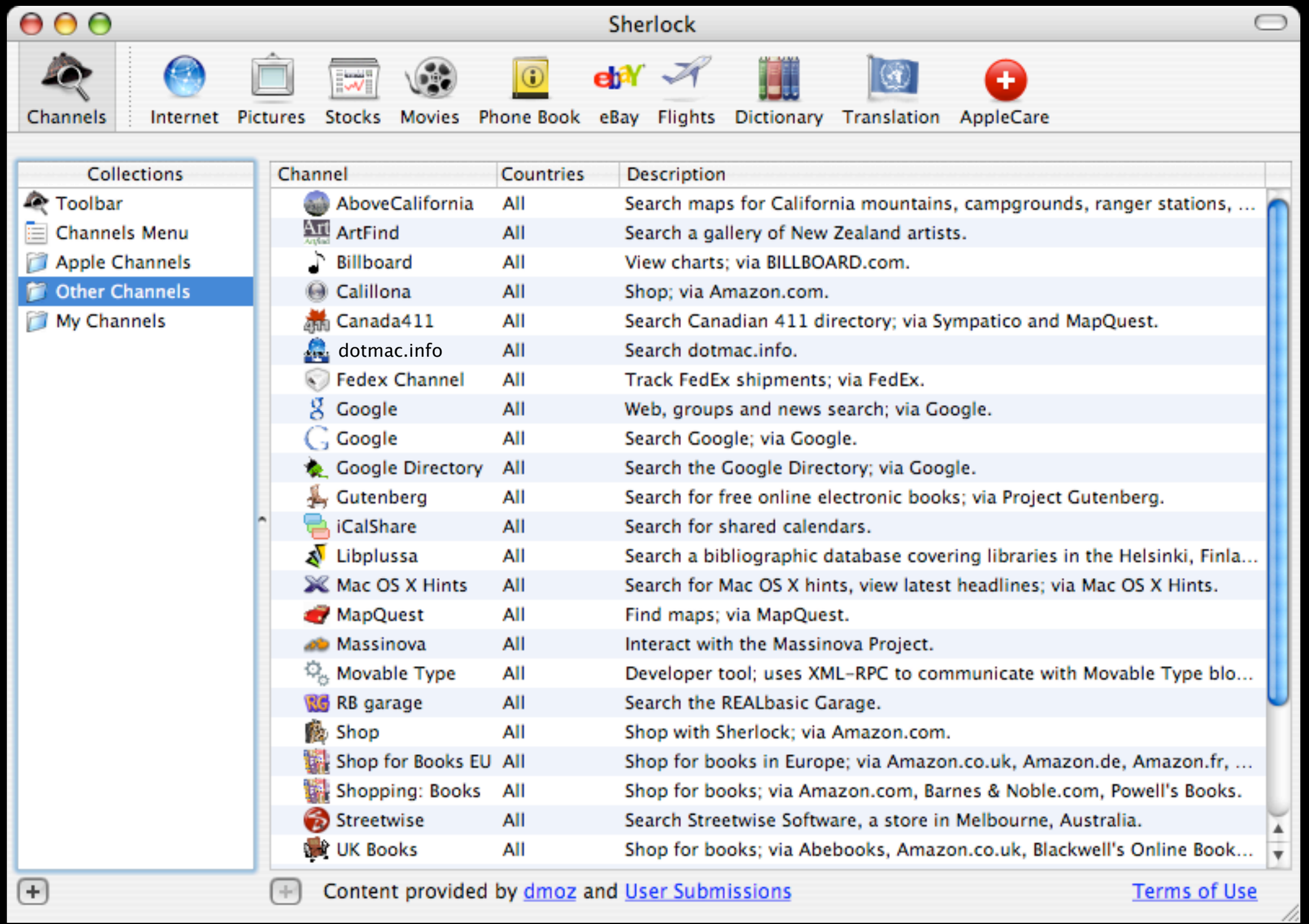
Channels View

Apple channels



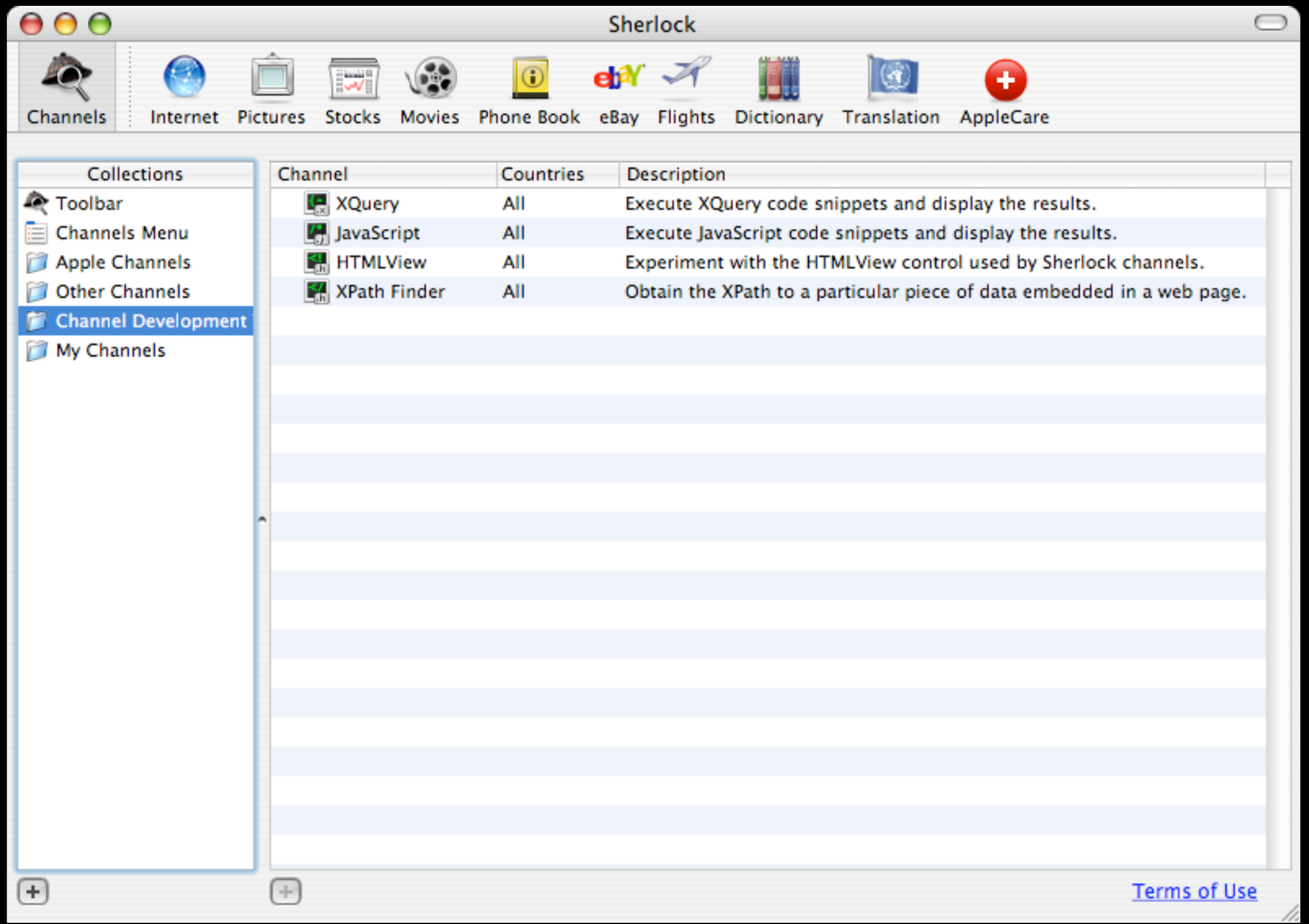
Channels View

Third Party channels



Channels View

Channel Development Tools



Demo

Jessica Kahn

Sherlock Channel Development Lead

Sherlock

- Web Services for the rest of us

Sherlock

- Web Services for the rest of us
- But what are Web Services?

Web Services

- SOAP?
- .NET?
- XML?
- WSDL?
- UDDI?
- ebXML?

Web Services

- Airline Reservations
- Movie Listings
- Yellow Pages
- Translating
- Shopping
- Finding

Delivering Web Services

- Challenges
  - Integration
    - Many sources
    - Frames vs. harnessing
  - User Interface
    - Native UI preferred
  - Change
  - Security

Sherlock Web Services

- Native user interface
- Quick to create
- Supports change on the web
- Integrates sources of information
- Secure Sandbox Environment
- Accessible via a Web Link or in Dock
- Third Party Channels listed in Sherlock

When to Create a Channel

- More than a web page
- Simpler than an application

	Web Page	Channel	Application
Quick to Create	✔	✔	
Change on the Web	✔	✔	
Sandbox Security	✔	✔	
Integrate Web Data		✔	
XML processing		✔	
Native Aqua UI		✔	✔
Launch from Dock	✔	✔	✔

Third Party Channels

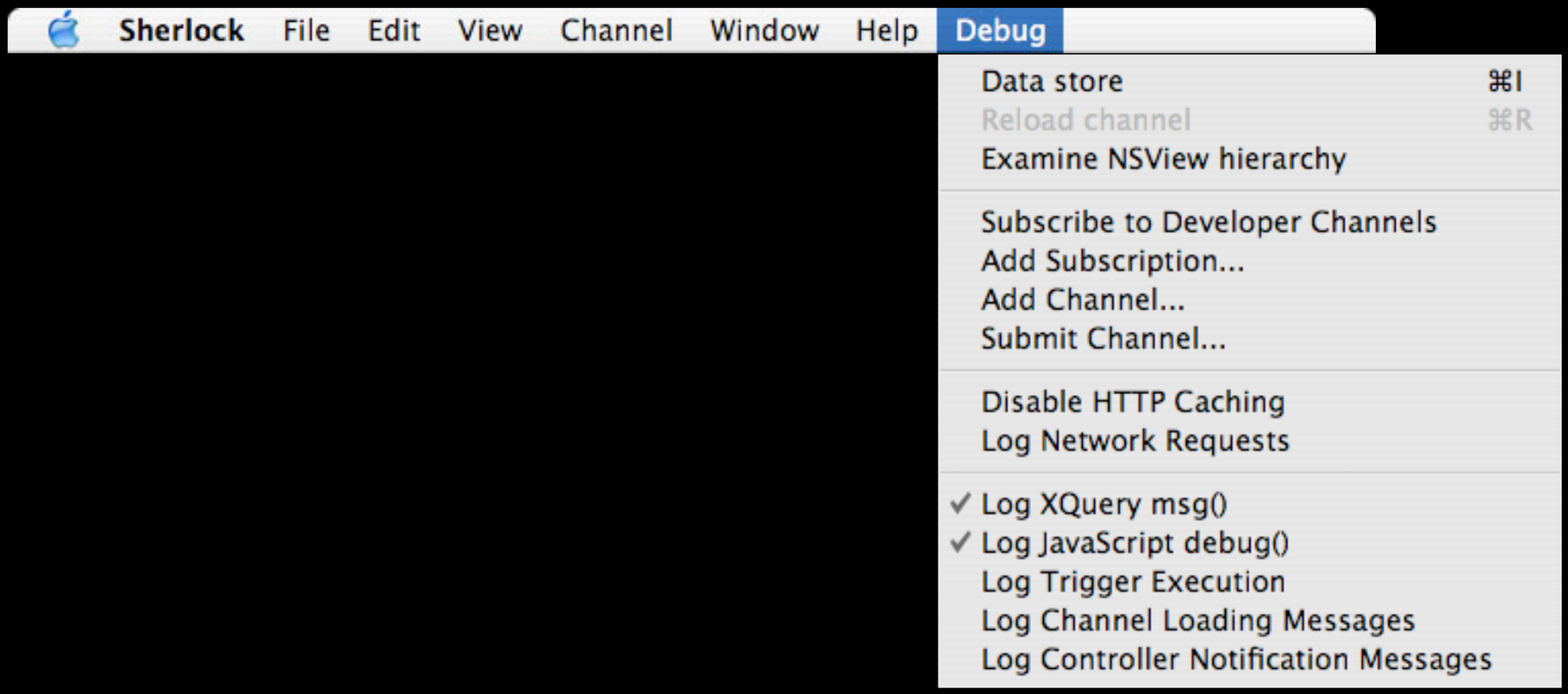
Demo
VersionTracker

Kevin Agren
Sr. Business Development Representative
TechTracker, Inc.

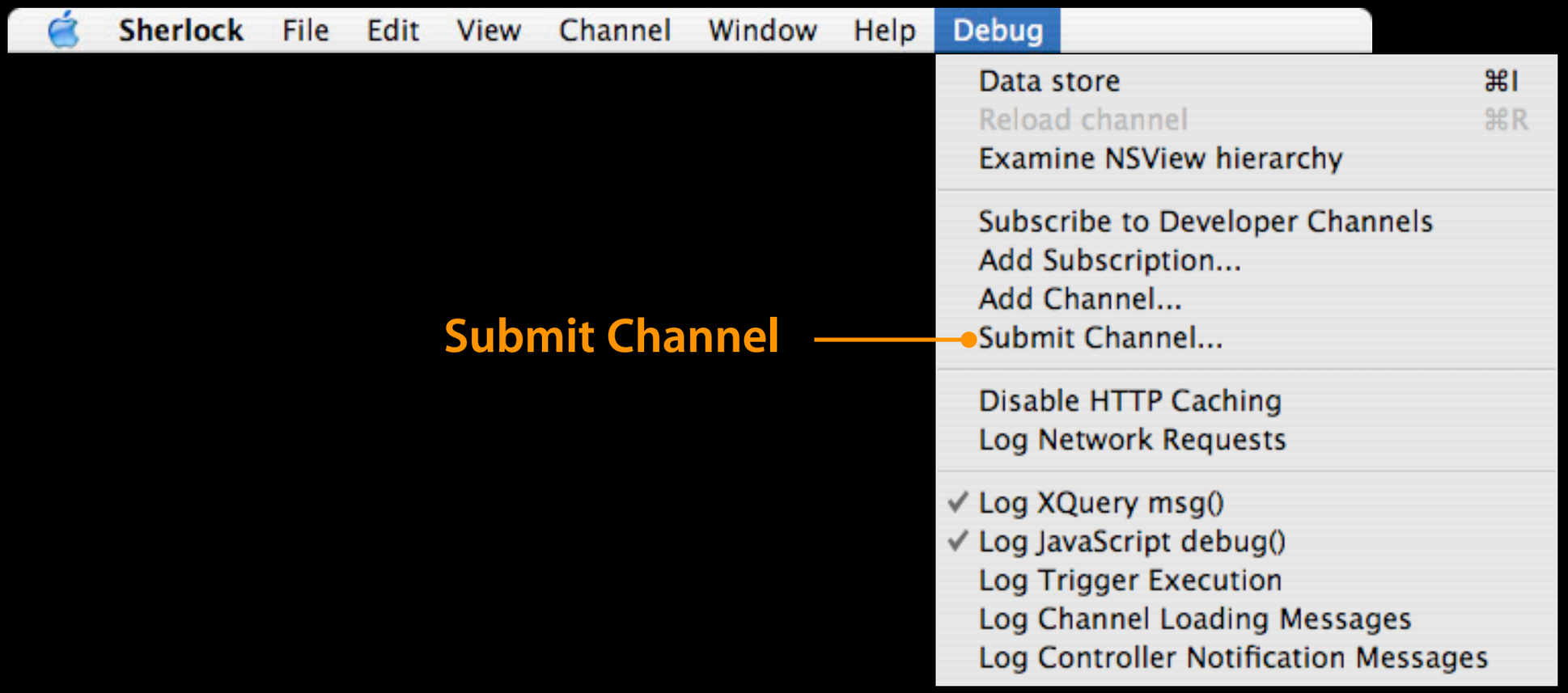
Survey

Developer Opportunity

- More compelling user interface
- Web link and update
- Comfortable for users
- Better visibility for your web services
- Direct access in Sherlock



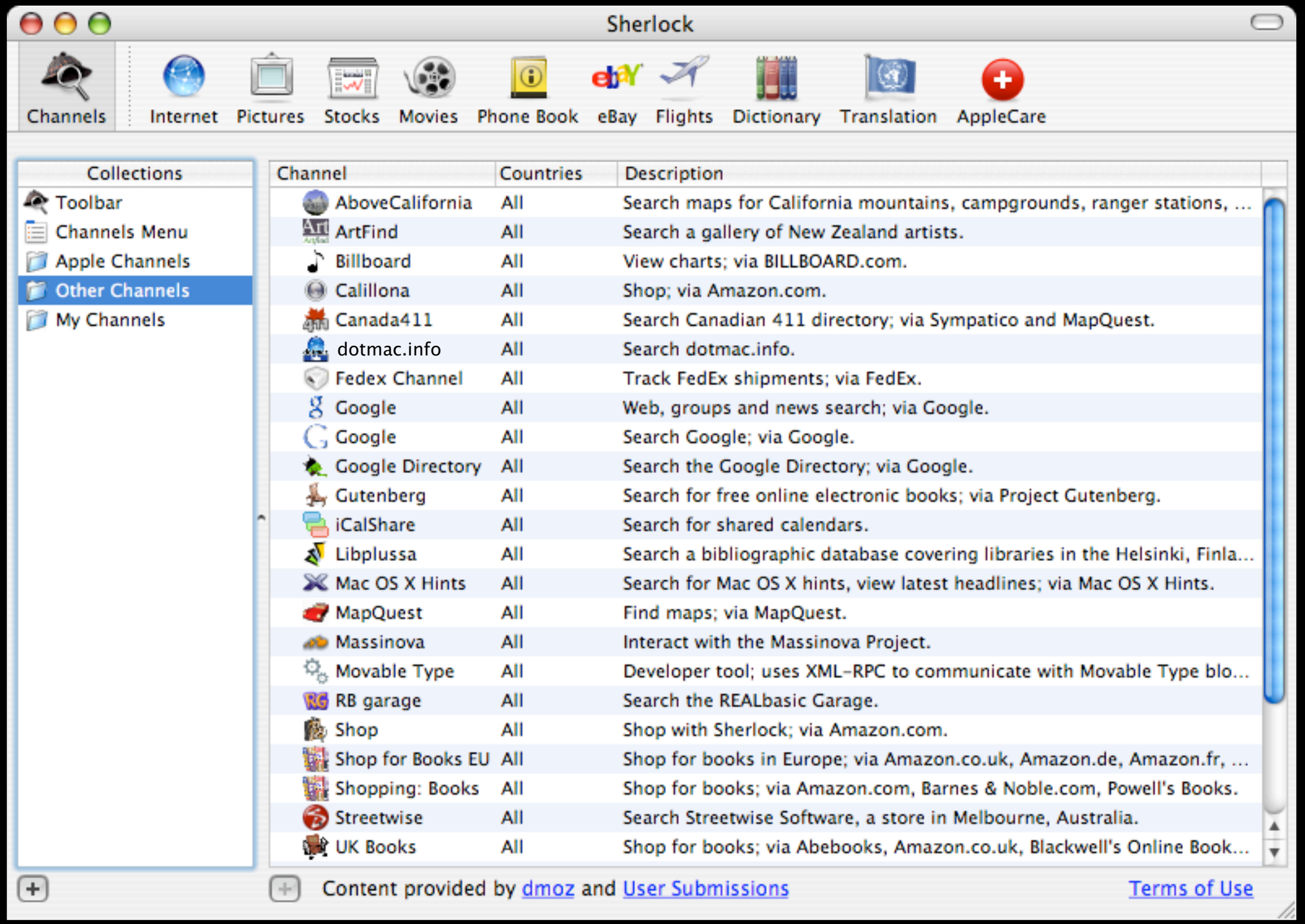
- Develop your channel



Submit Channel

- Develop your channel
- Submit to Apple

Built-In to Sherlock



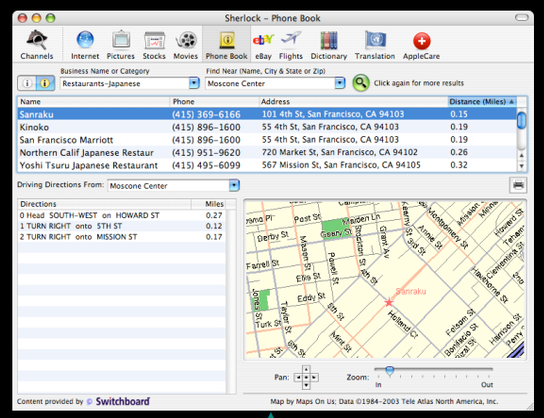
Structure of a Sherlock Channel

- Channel design principles
- Key elements of a channel
- Demo of XQuery
- Sherlock Extensions

Channel Design Principles

- Focus on one topic
- Start with information showing
- Query the Web only when necessary
- Remember user state
- Follow Aqua guidelines

Key Elements of a Channel



DataStore

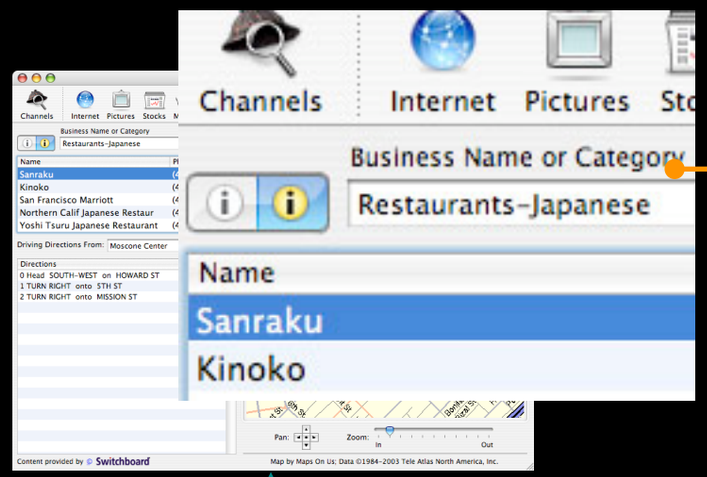
Triggers (JavaScript)

Services (XQuery)

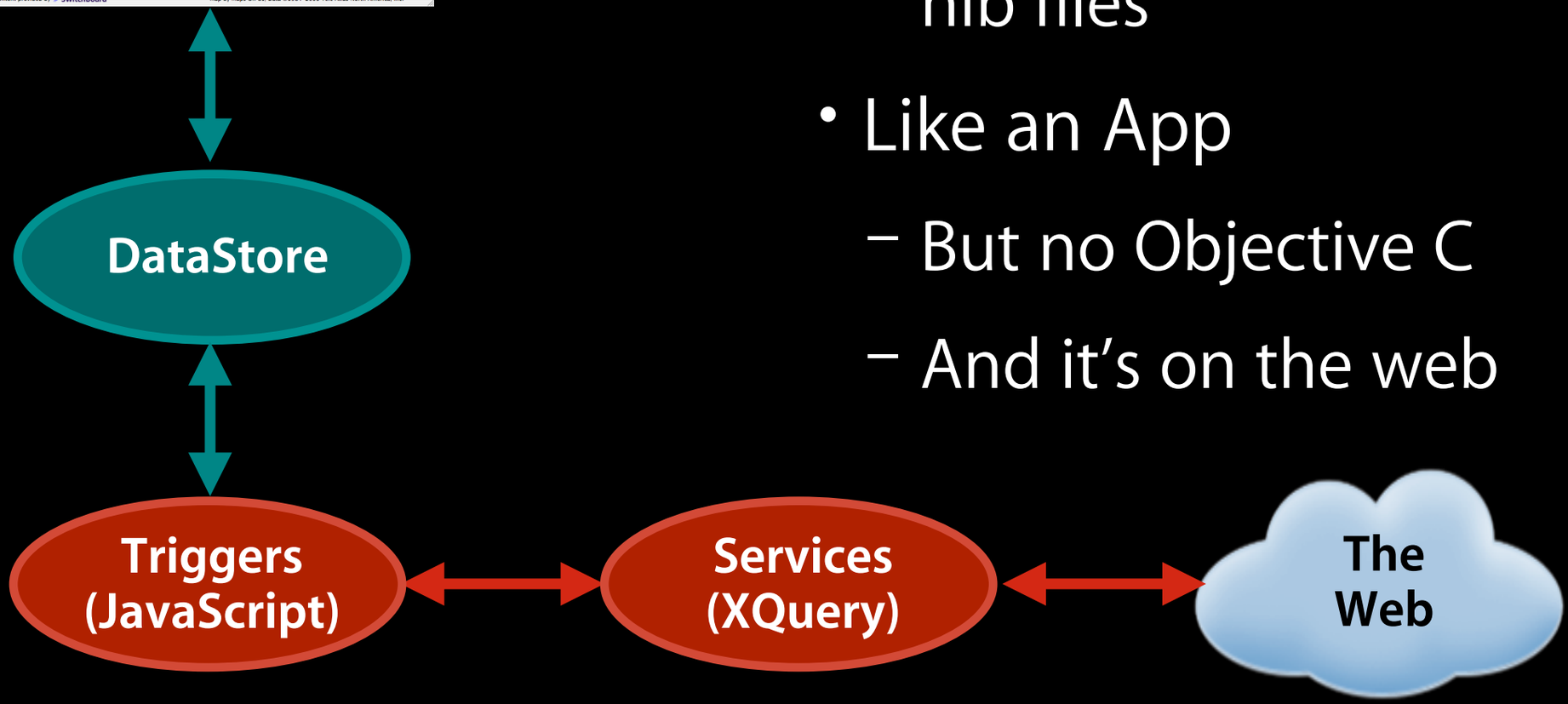
The Web

Key Elements of a Channel

User Interface

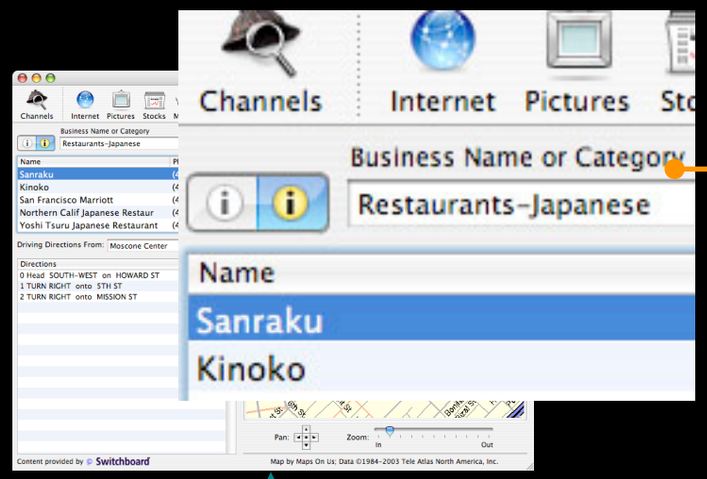


- User Interface
  - Interface Builder
  - nib files
- Like an App
  - But no Objective C
  - And it's on the web



Key Elements of a Channel

User Interface



- Name UI Elements
  - PhoneBook.QueryField
  - PhoneBook.Results

DataStore

Triggers (JavaScript)

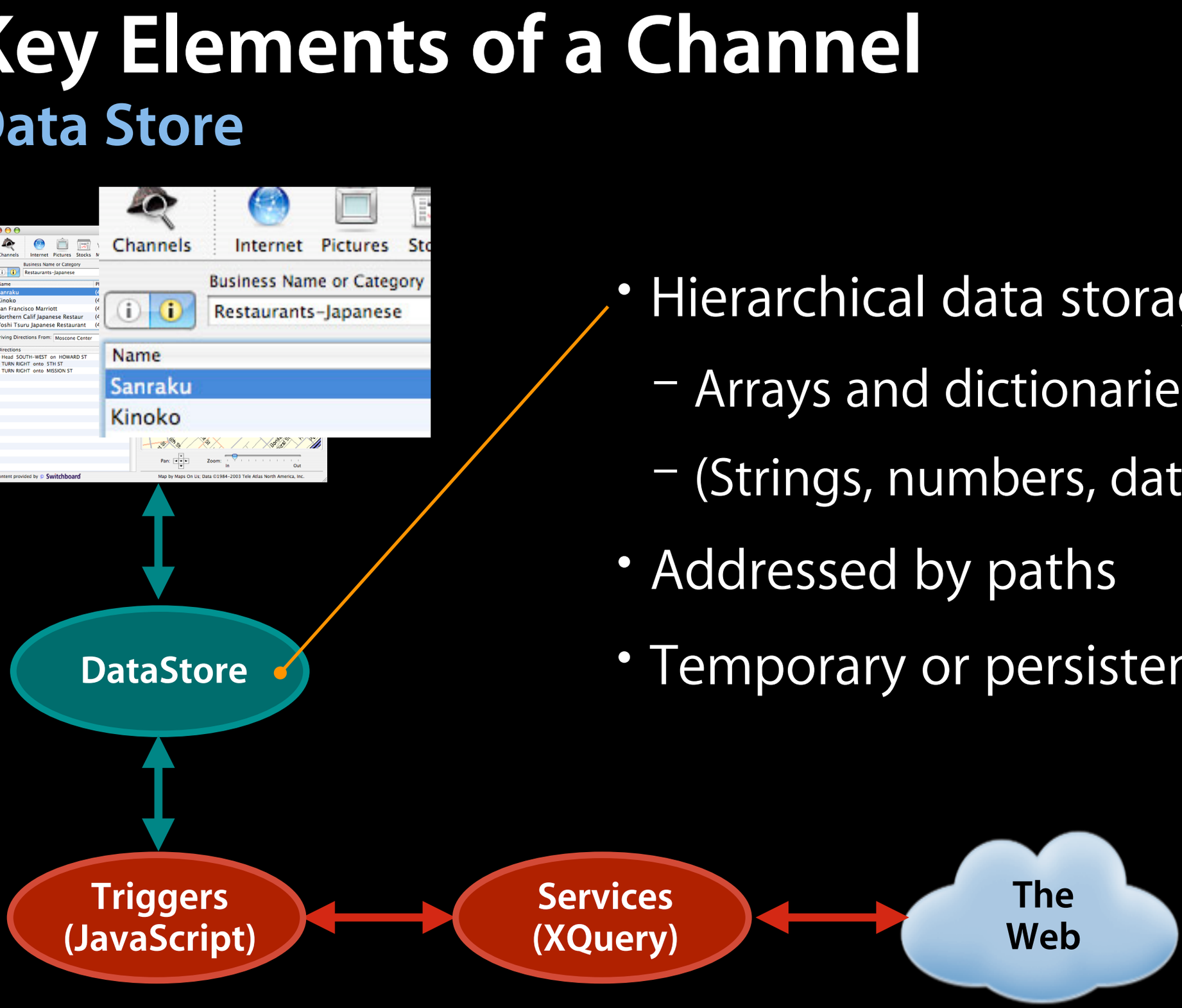
Services (XQuery)

The Web

Key Elements of a Channel

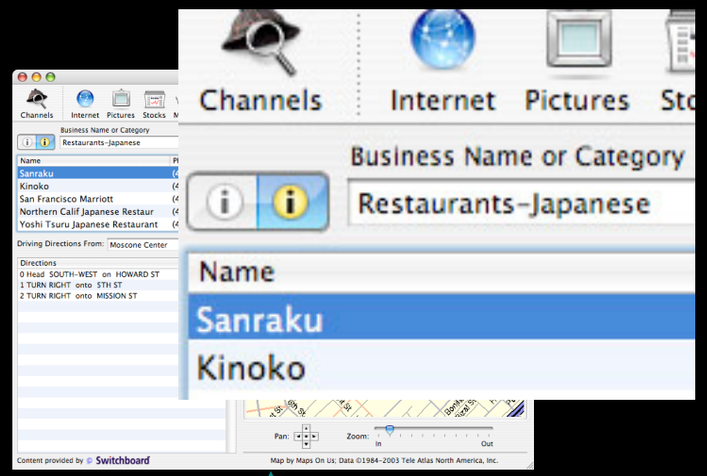
Data Store

- Hierarchical data storage
  - Arrays and dictionaries
  - (Strings, numbers, data)
- Addressed by paths
- Temporary or persistent

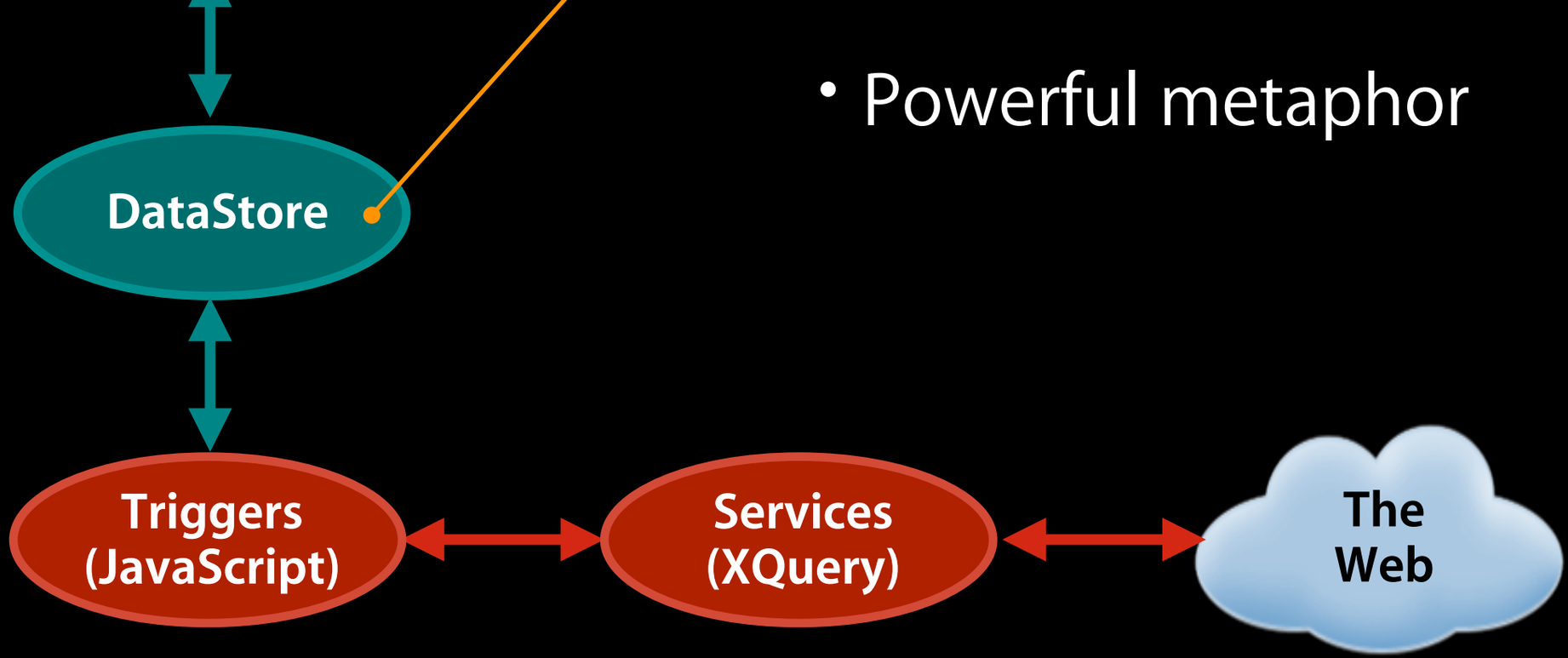


Key Elements of a Channel

Data Store

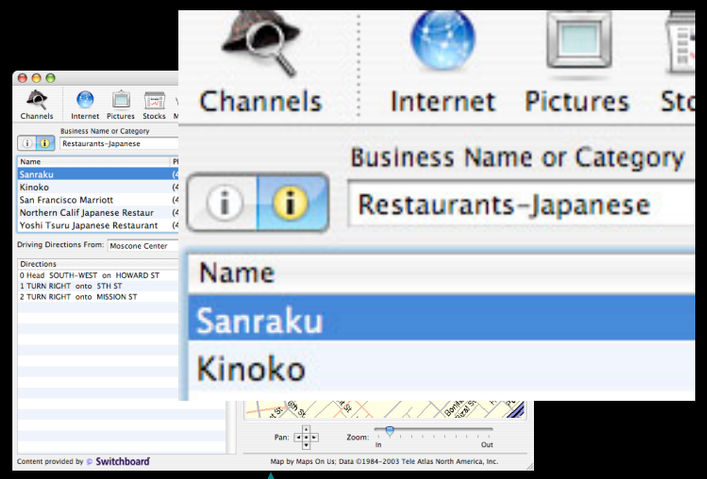


- Always reflects the UI
  - Read -> see current value
  - Write -> change the UI
- Powerful metaphor

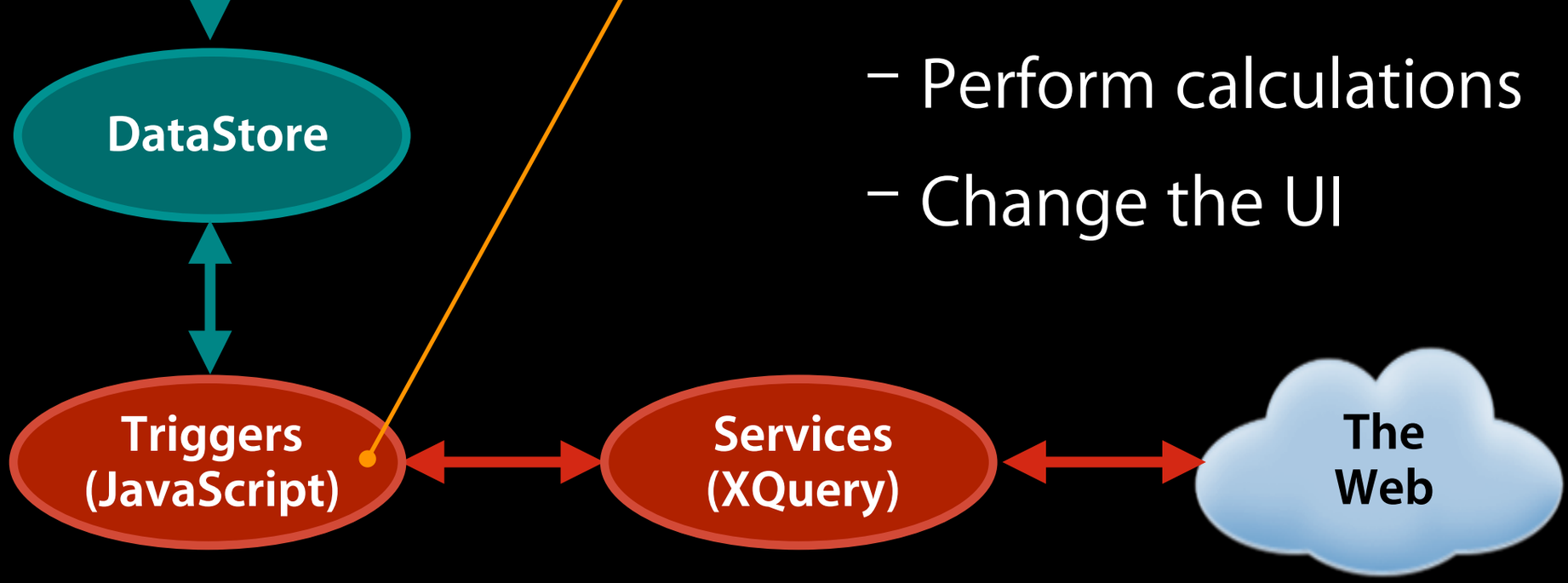


Key Elements of a Channel

Triggers (JavaScript)

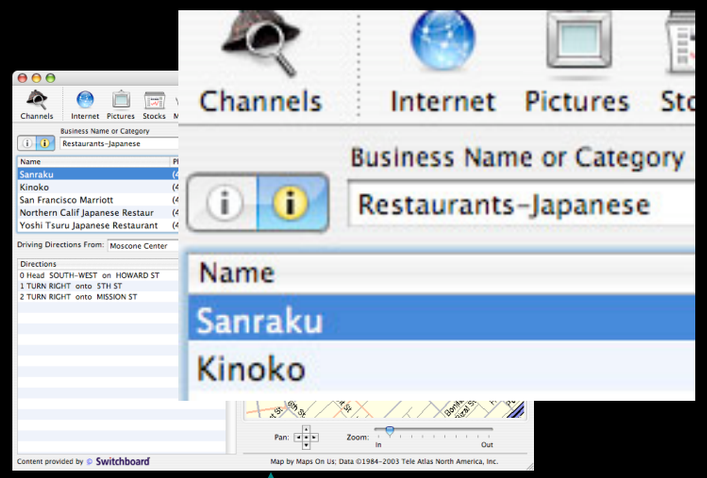


- Make things happen
  - Run when Data Store changes
  - Invoke Services
  - Perform calculations
  - Change the UI

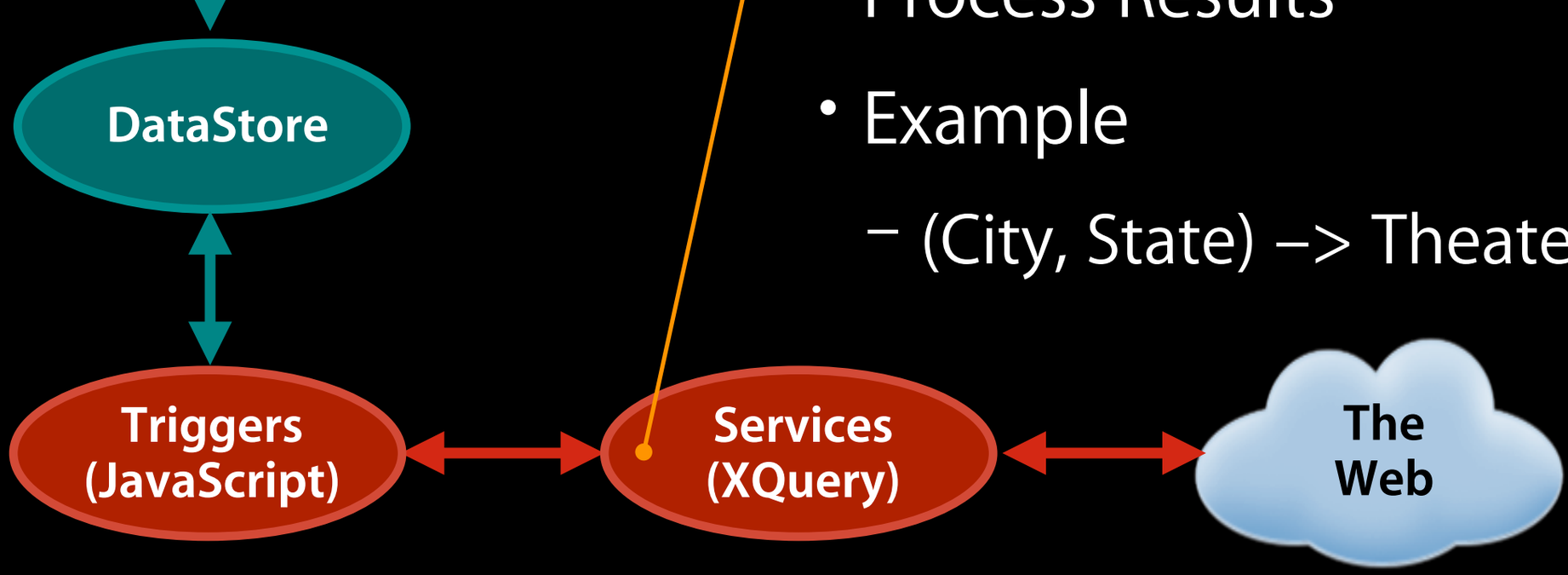


Key Elements of a Channel

Services (XQuery)

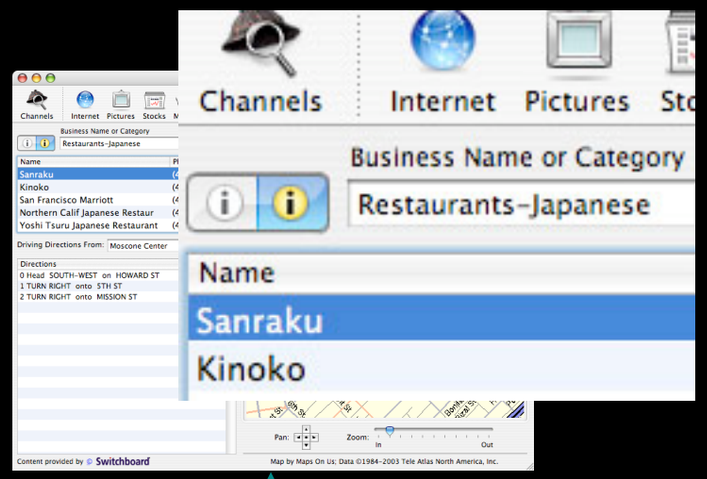


- Query the Web
  - A web page
  - A web service
- Process Results
- Example
  - (City, State) –> Theaters

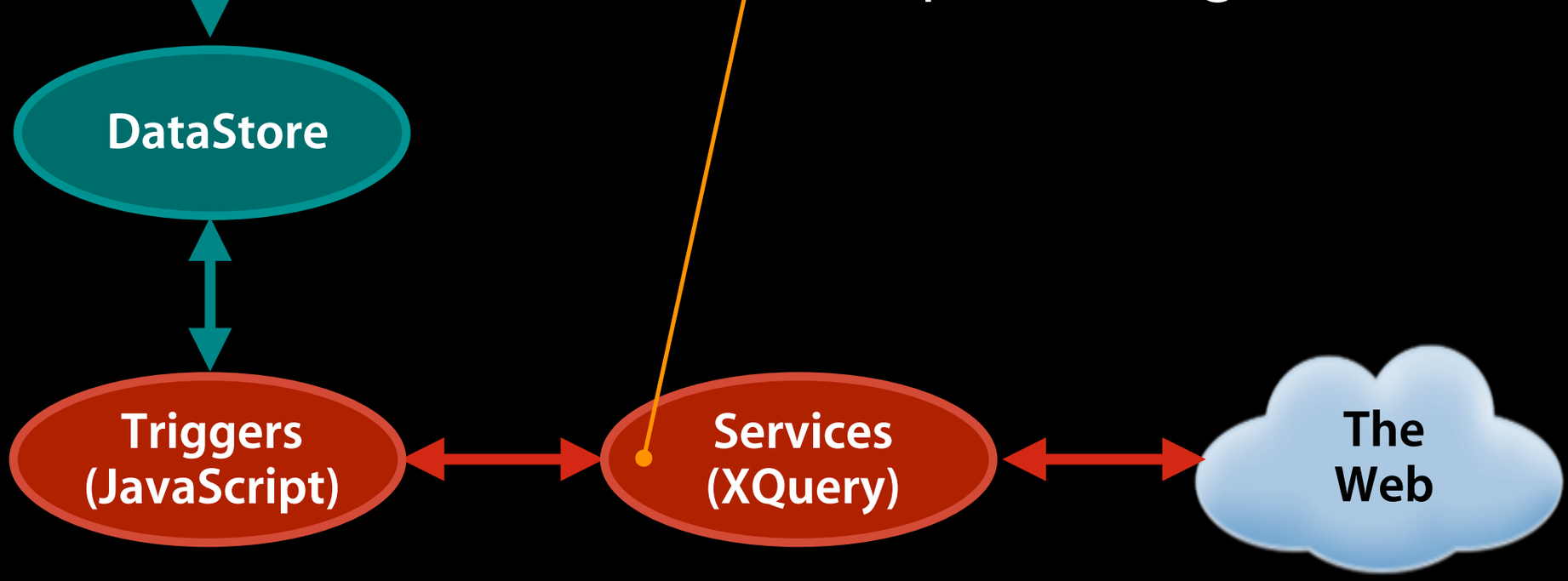


Key Elements of a Channel

Services (XQuery)



- XQuery
  - W3C draft spec
  - Designed for processing XML



Demo

XQuery and XPath Finder

Sherlock Extensions

- JavaScript
  - DataStore
  - System
  - XMLQuery
- XQuery
  - http support
  - URL encoding
  - Data matching

Channel Development How To

Jessica Kahn
Sherlock Channel Development Lead

Channel Development How To

- Developing channels in four steps
- What's new for developers
- Advanced topics

Developing Channels In Four Steps

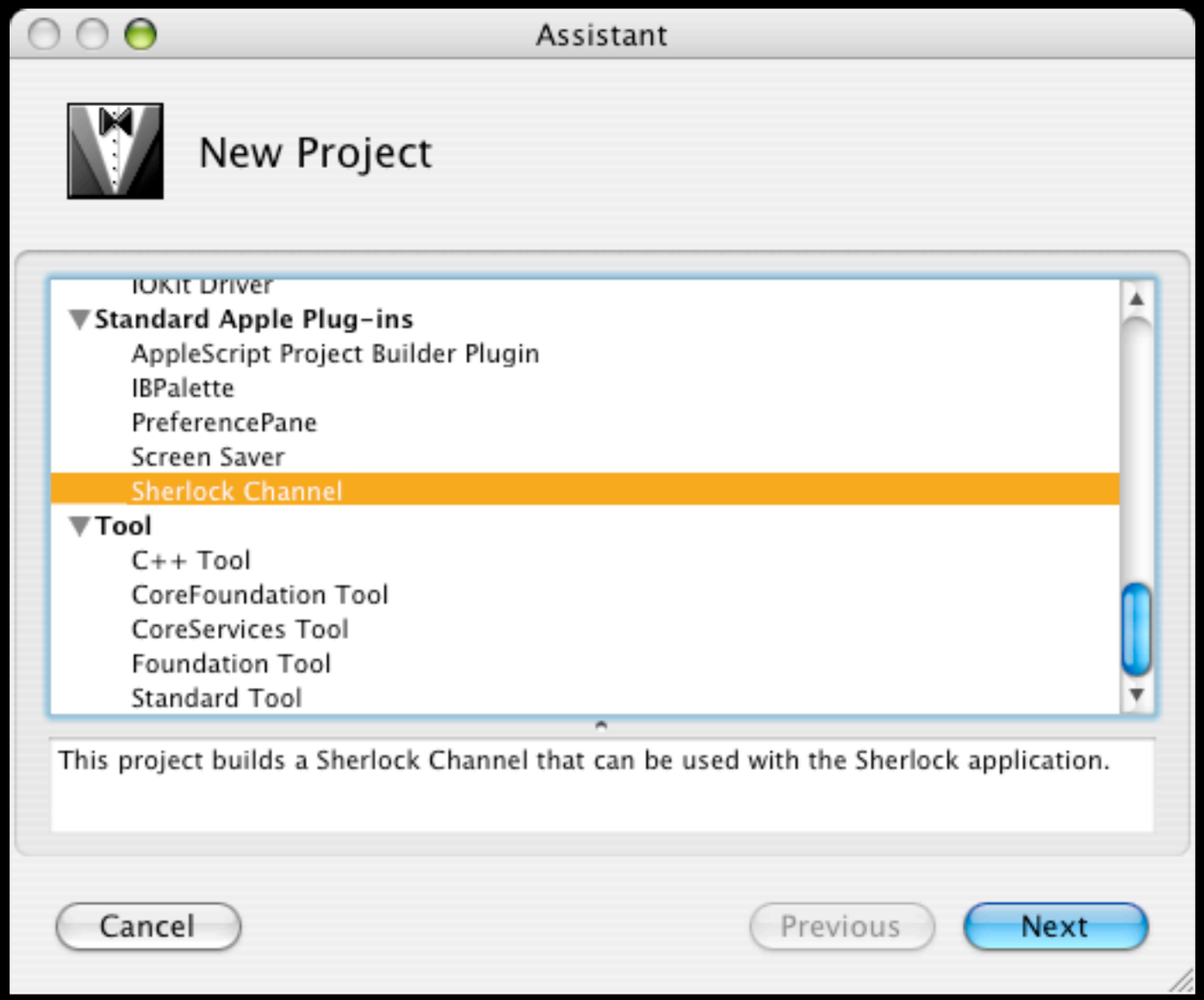
1. Create
2. Customize
3. Debug
4. Deploy

Developing Channels In Four Steps

1. Create
2. Customize
3. Debug
4. Deploy

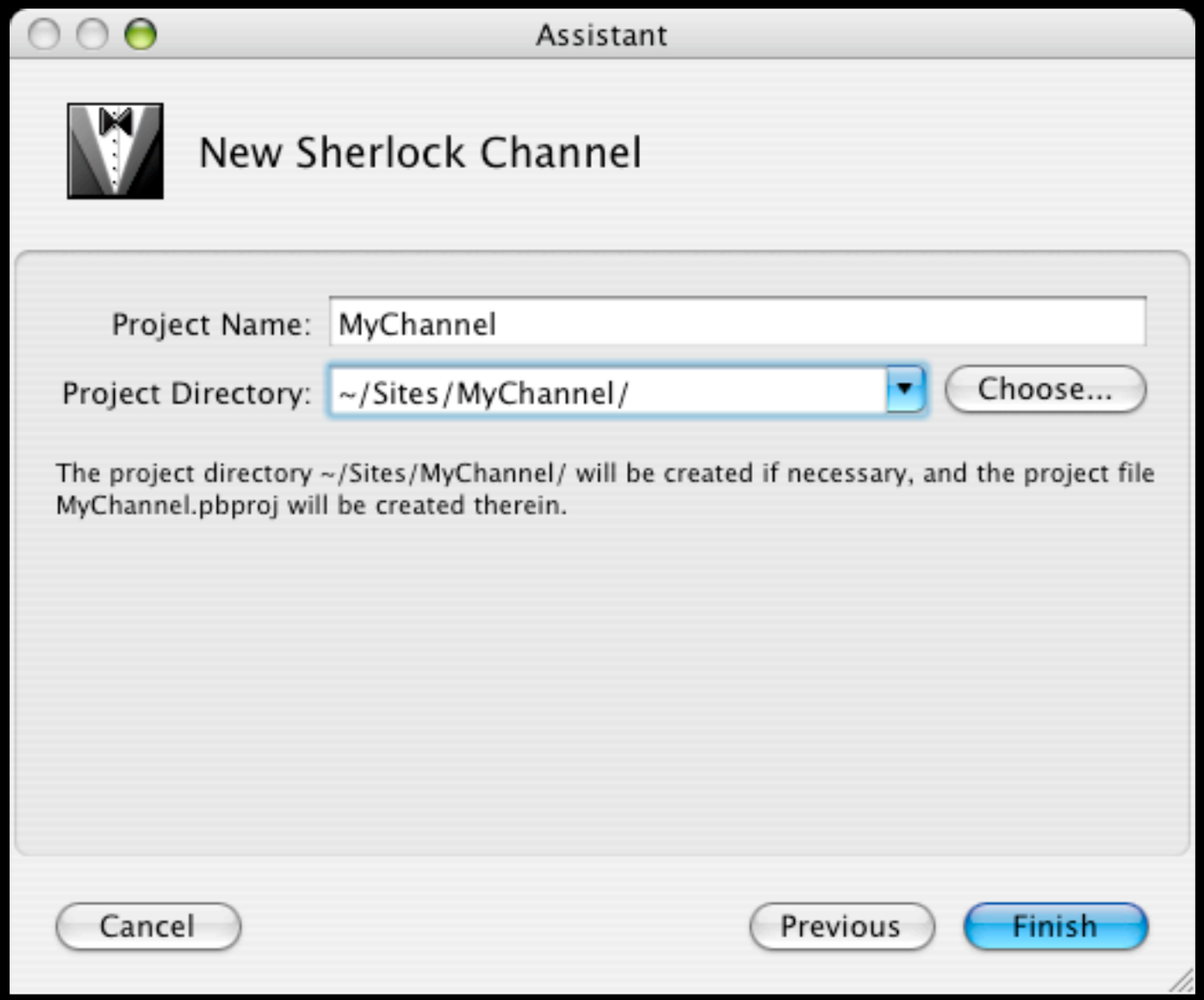
Create

Use the Sherlock Channel project template



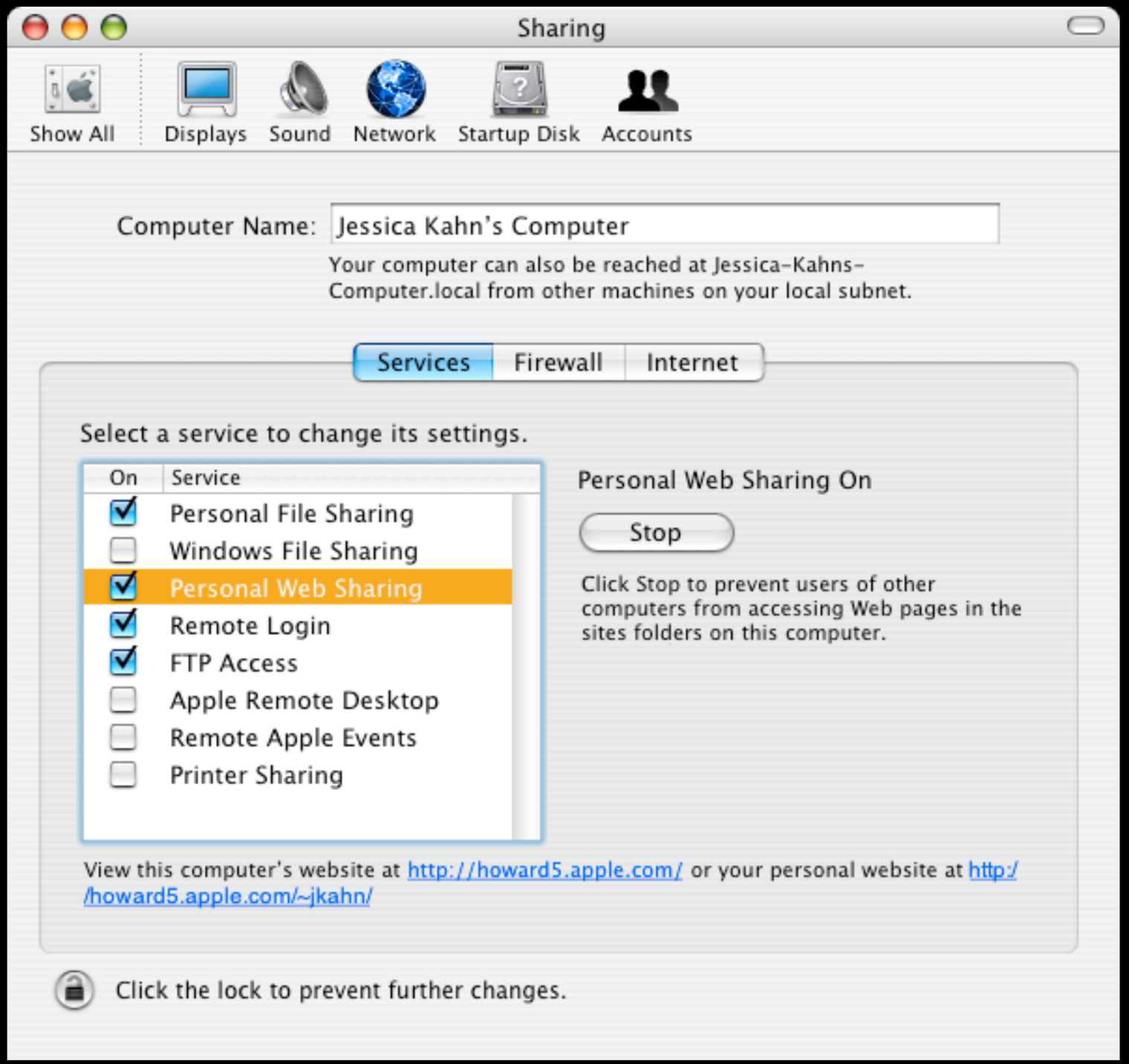
Create

Save your project in an http-accessible directory



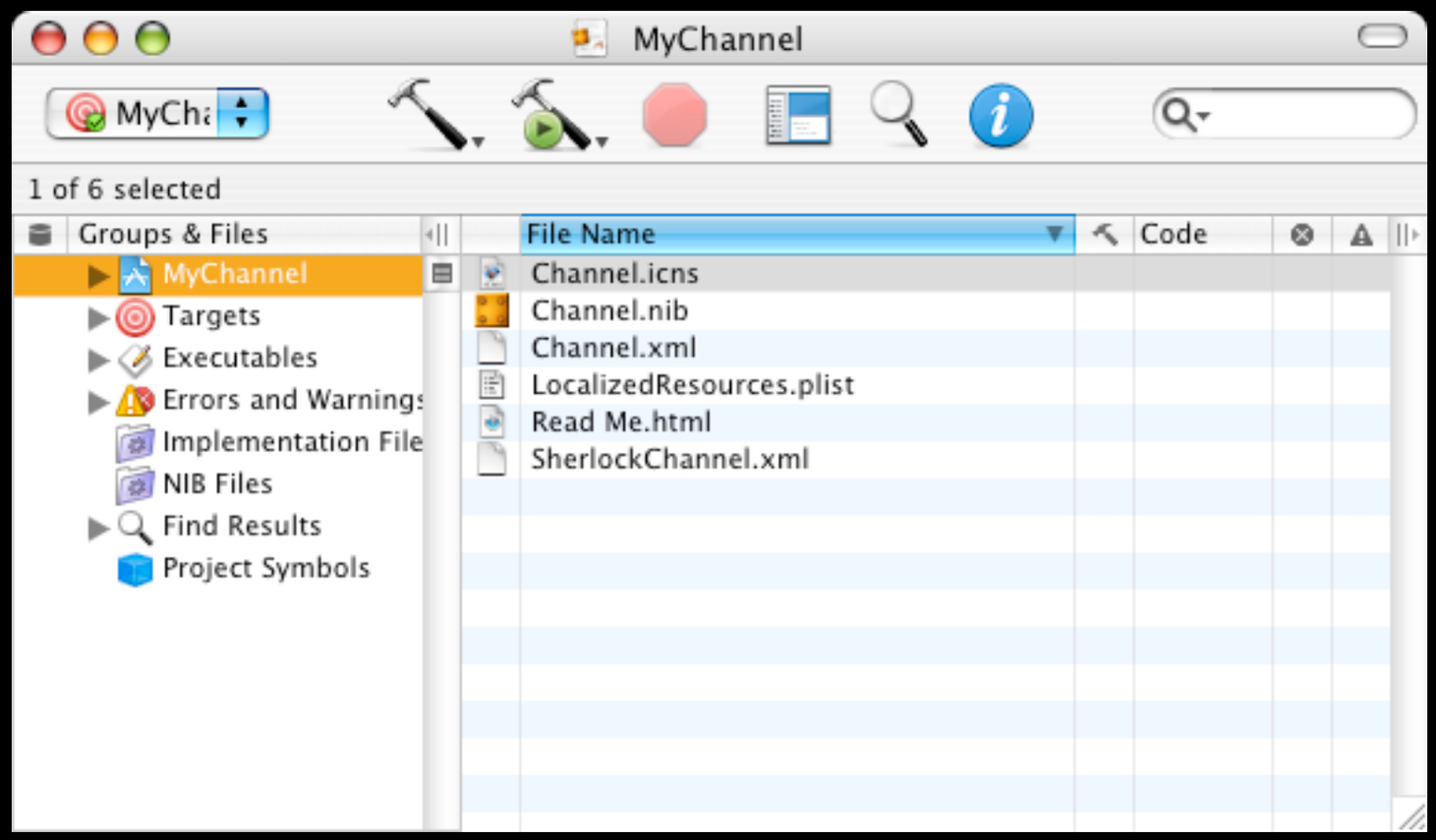
Create

Turn on Personal Web Sharing



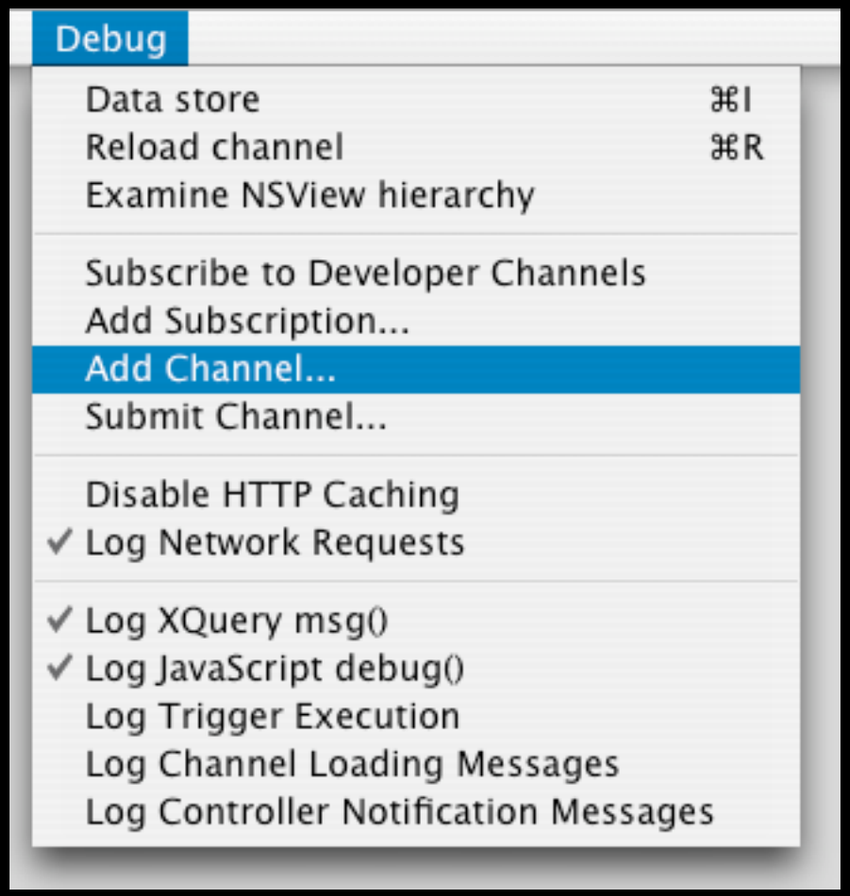
Create

Build and run your project



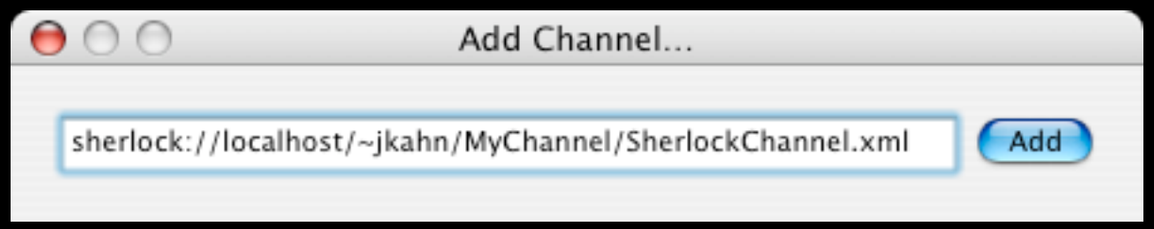
Create

Add your channel to Sherlock



Create

Add your channel to Sherlock



Create

Add your channel to Sherlock

Developing Channels In Four Steps

1. Create
2. Customize
3. Debug
4. Deploy

Customize

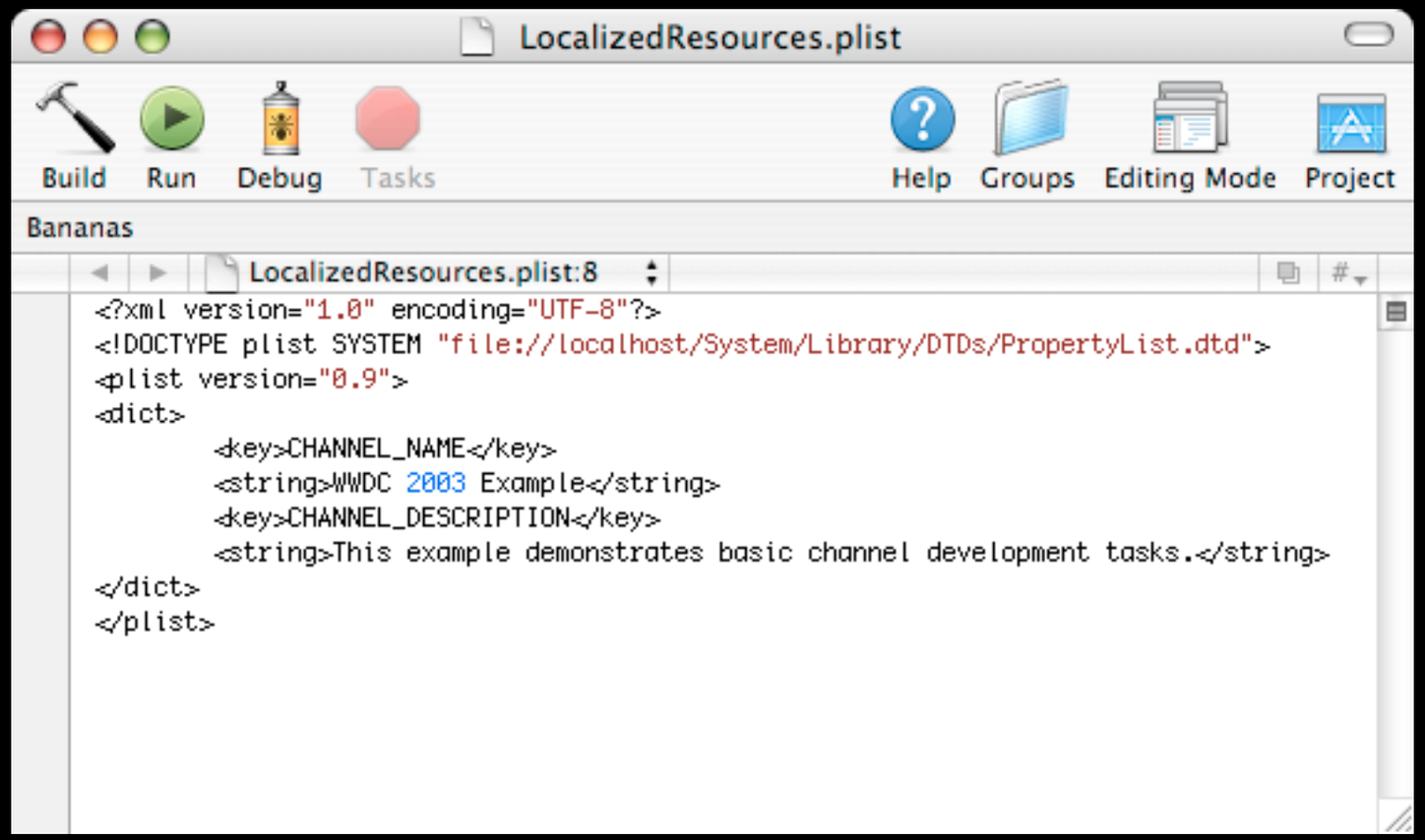
- Give your channel its own identity
- Enhance starter code to meet your needs

Give Your Channel Its Own Identity

- Name it
- Describe it
- Set its identifier
- Give it a beautiful icon

Give Your Channel Its Own Identity

Name it



Give Your Channel Its Own Identity

Name it

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist SYSTEM "file://localhost/System/Library
   DTDs/PropertyList.dtd">
<plist version="0.9">
<dict>
   <key>CHANNEL_NAME</key>
   <string>WWDC 2003 Example</string>
   <key>CHANNEL_DESCRIPTION</key>
   <string>This example demonstrates basic channel
      development tasks.</string>
</dict>
</plist>
```

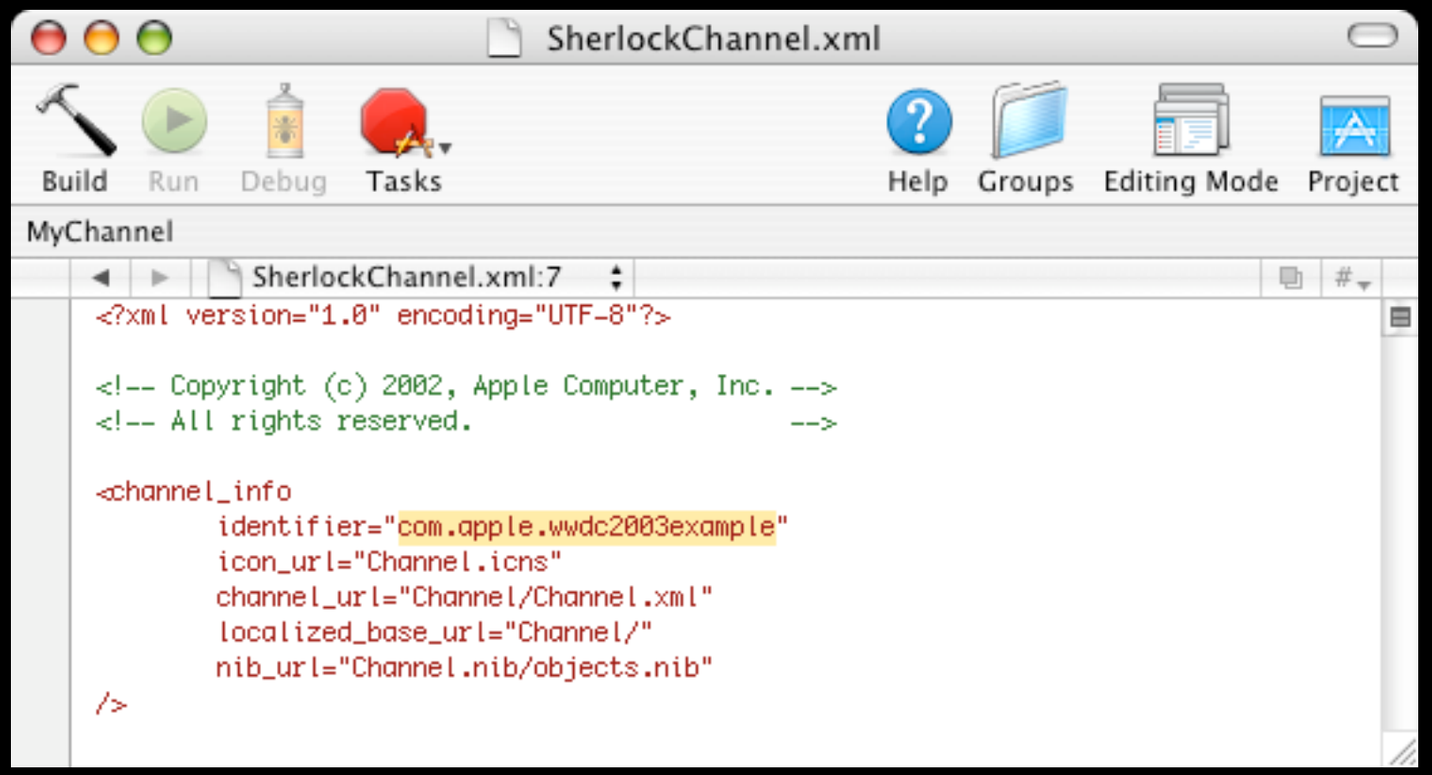
Give Your Channel Its Own Identity

Describe it

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist SYSTEM "file://localhost/System/Library
   DTDs/PropertyList.dtd">
<plist version="0.9">
<dict>
   <key>CHANNEL_NAME</key>
   <string>WWDC 2003 Example</string>
   <key>CHANNEL_DESCRIPTION</key>
   <string>This example demonstrates basic channel
      development tasks.</string>
</dict>
</plist>
```

Give Your Channel Its Own Identity

Set its identifier



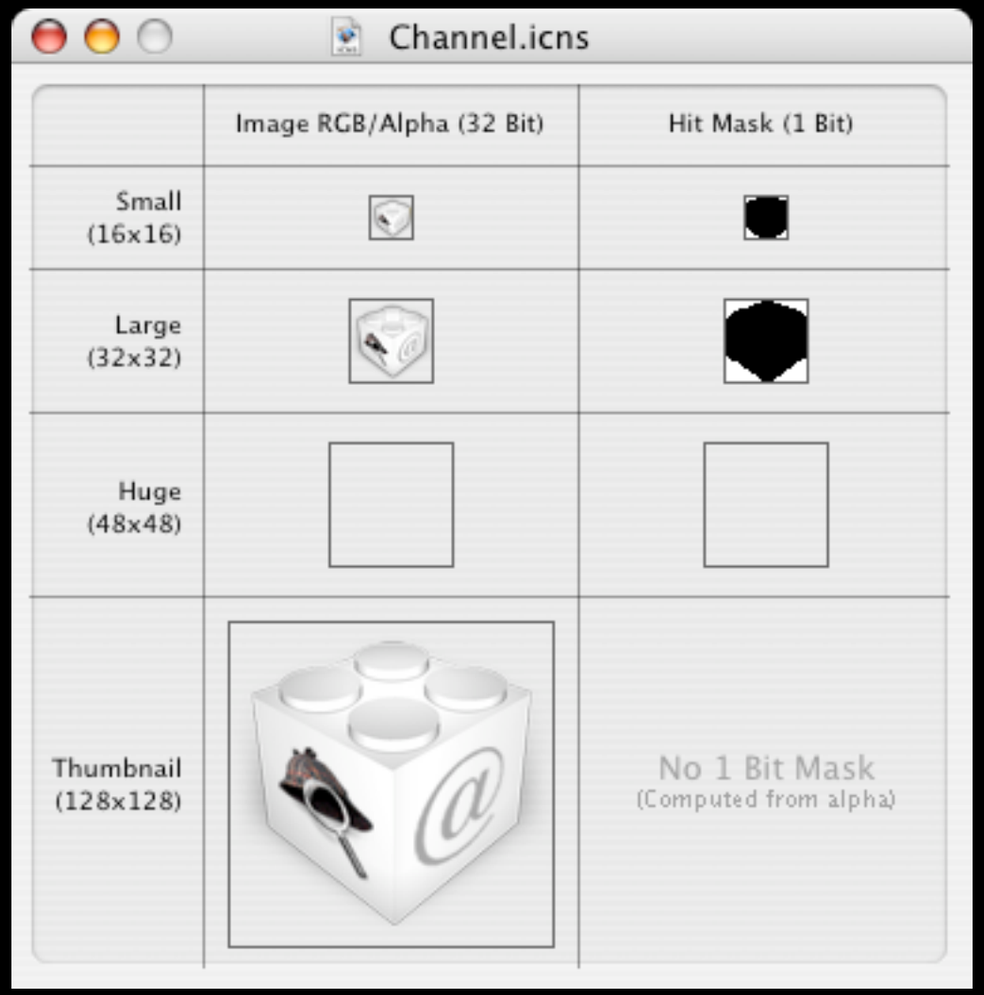
Give Your Channel Its Own Identity

Set its identifier

```
<?xml version="1.0" encoding="UTF-8"?>
<channel_info
   identifier="com.apple.wwdc2003example"
   icon_url="Channel.icns"
   channel_url="Channel/Channel.xml"
   localized_base_url="Channel/"
   nib_url="Channel.nib/objects.nib"
/>
```

Give Your Channel Its Own Identity

Give it a beautiful icon



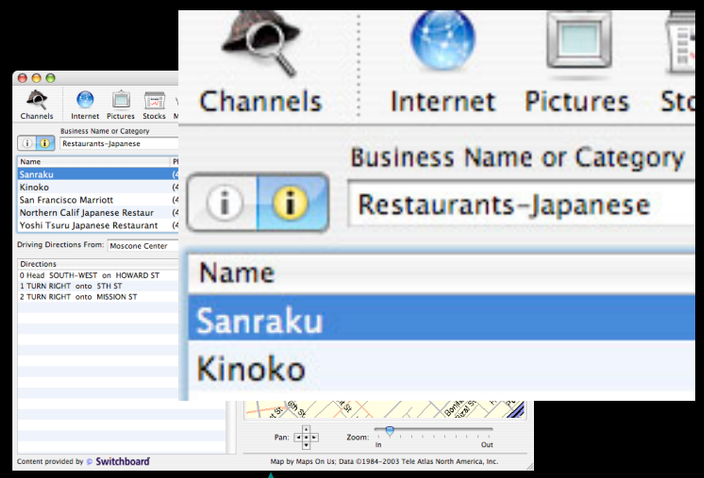
Give It a Beautiful Icon

- Get the Aqua Icon Kit, or
- Hire a professional design agency
- http://developer.apple.com/ue/aqua/icons.html

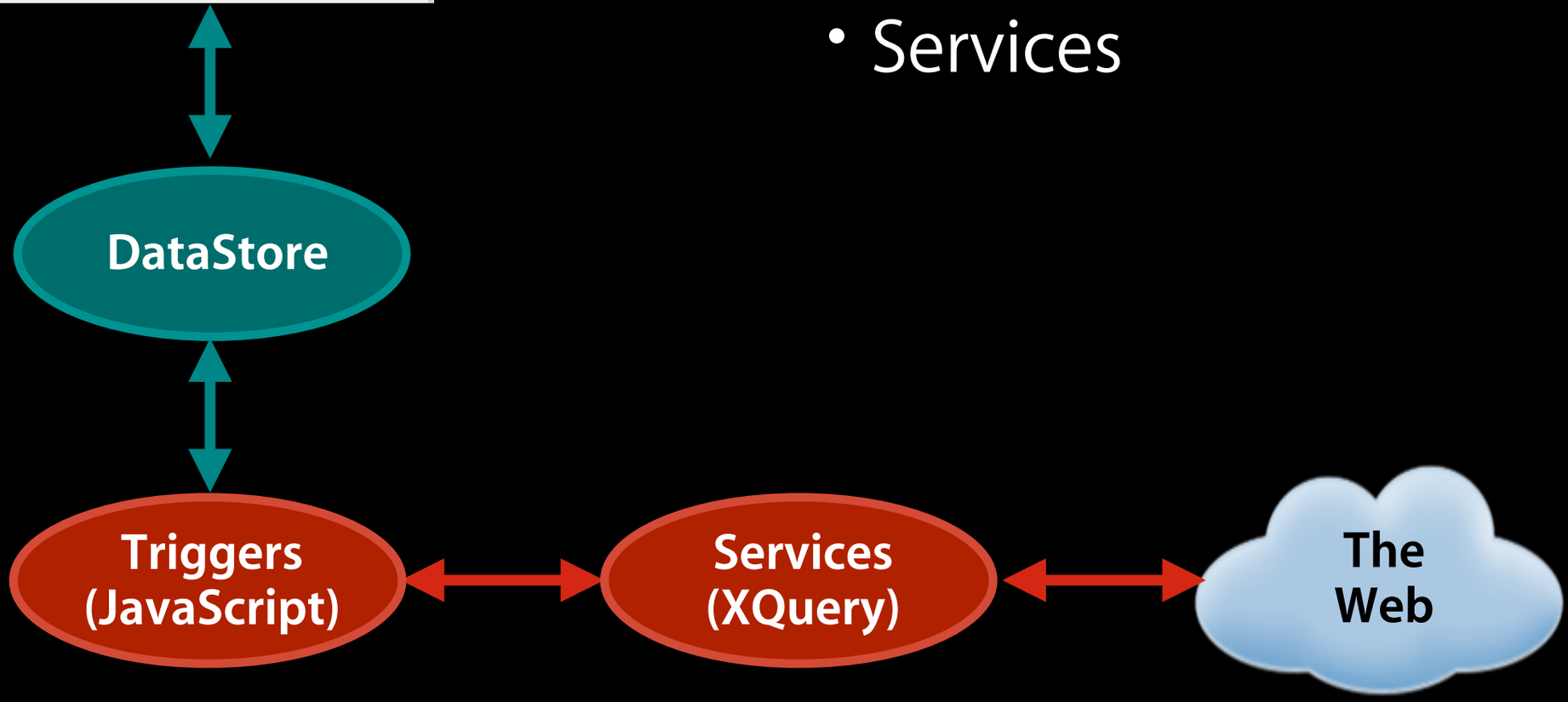
Customize

- Give your channel its own identity
- Enhance starter code to meet your needs

Enhance Code to Meet Your Needs

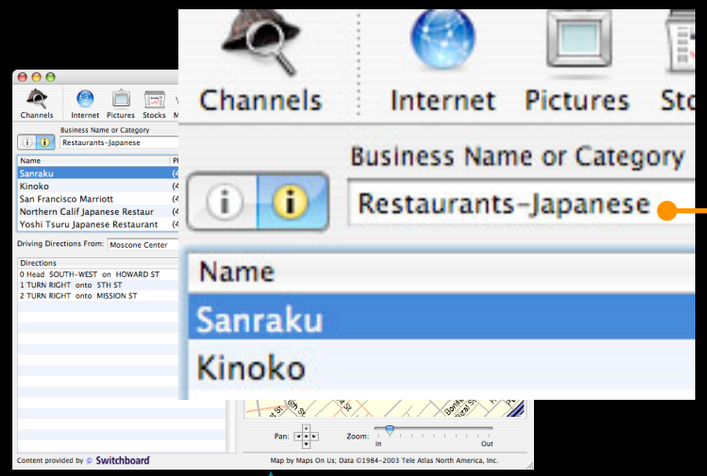


- Interface
- Triggers
- Services

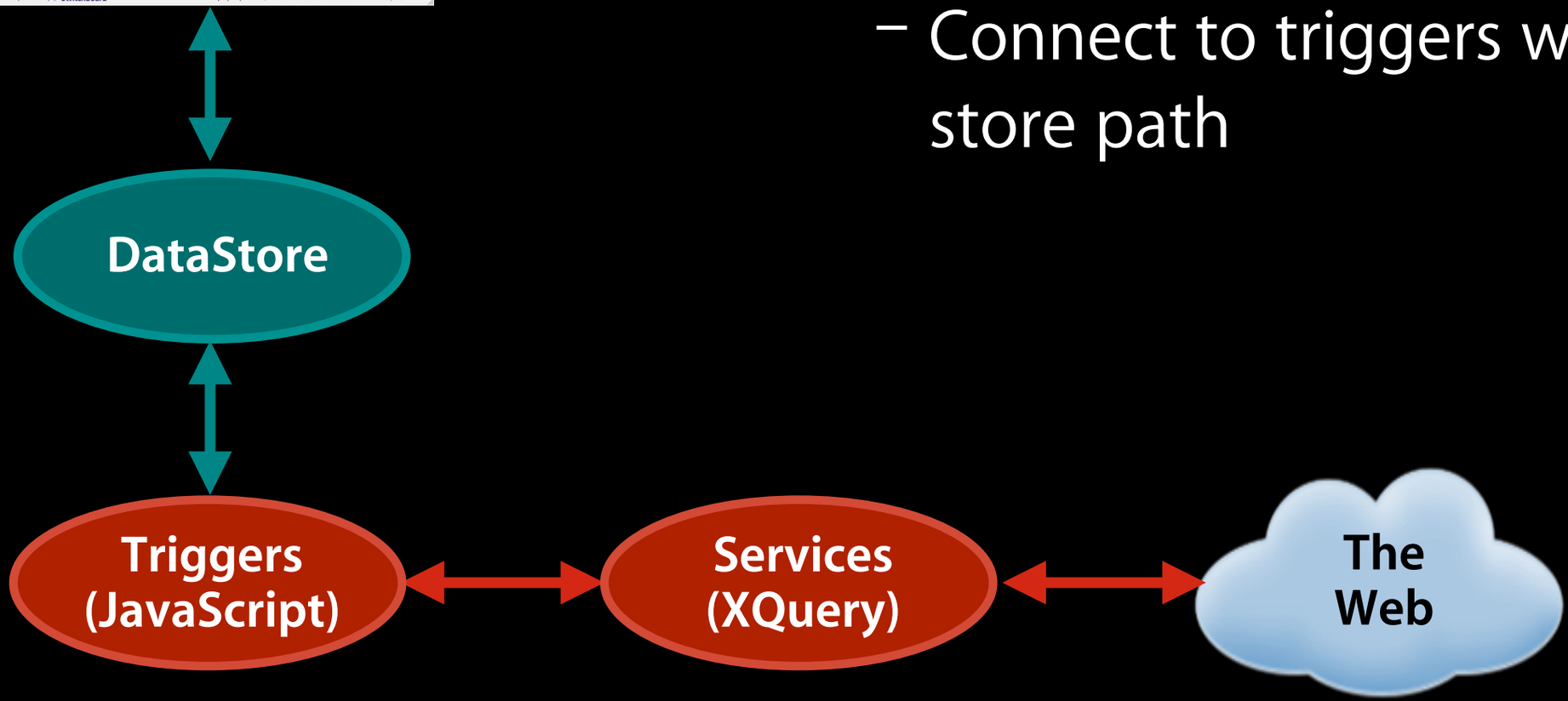


Enhance Code to Meet Your Needs

Interface

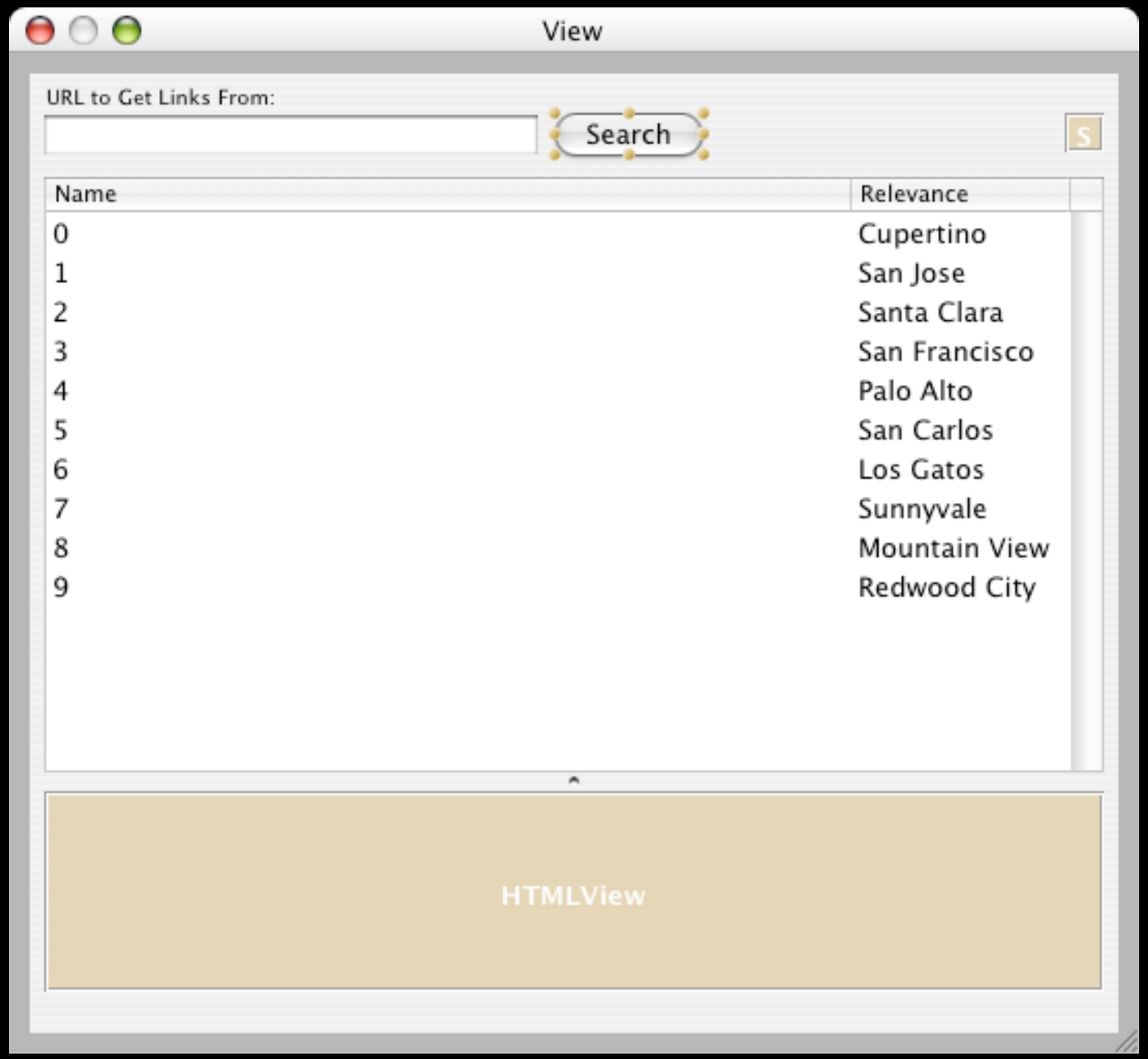


- Interface
  - Design with Interface Builder
  - Connect to triggers with data store path



Interface

Design with Interface Builder



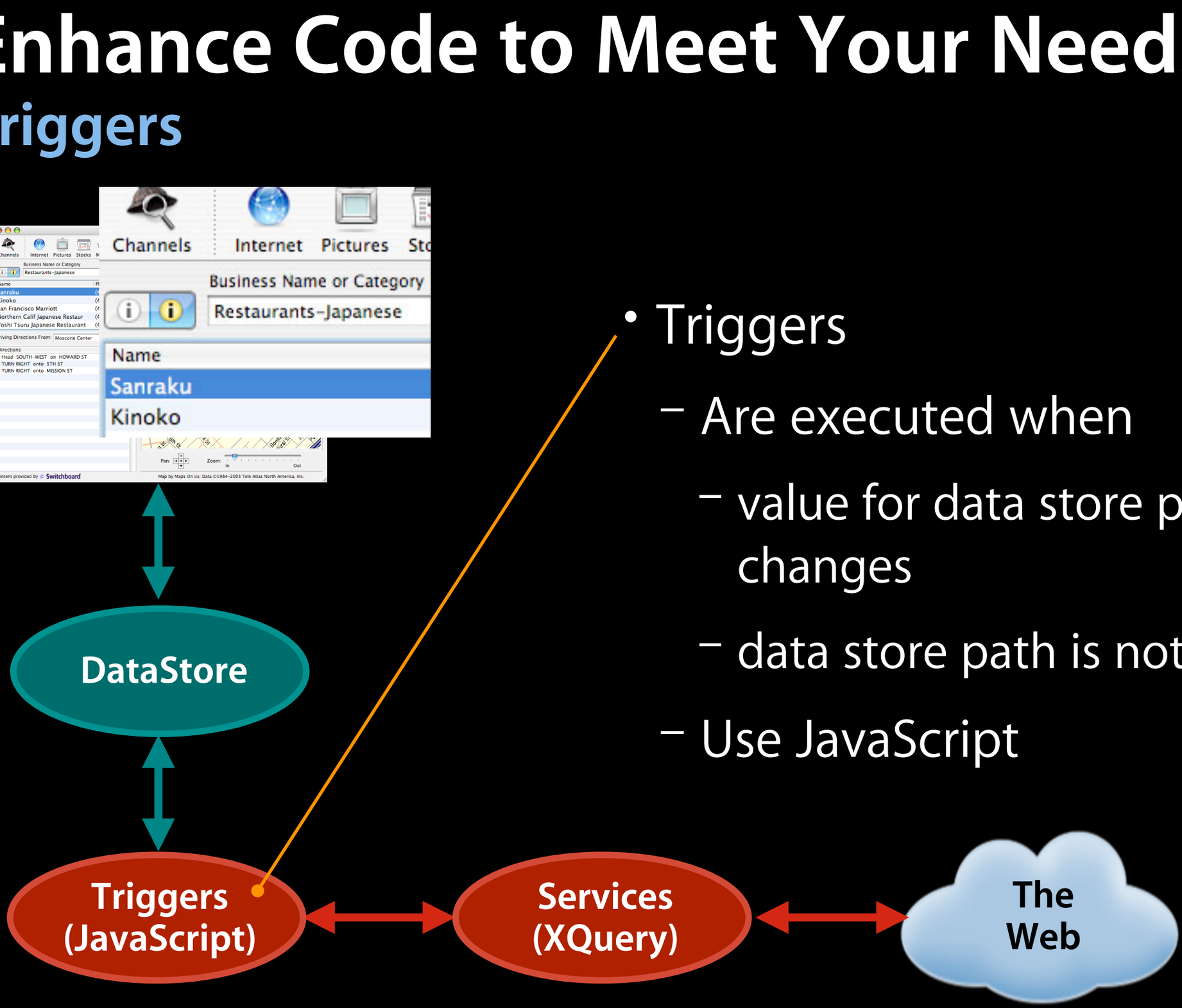
Interface and Triggers

Connect by setting Sherlock data store path

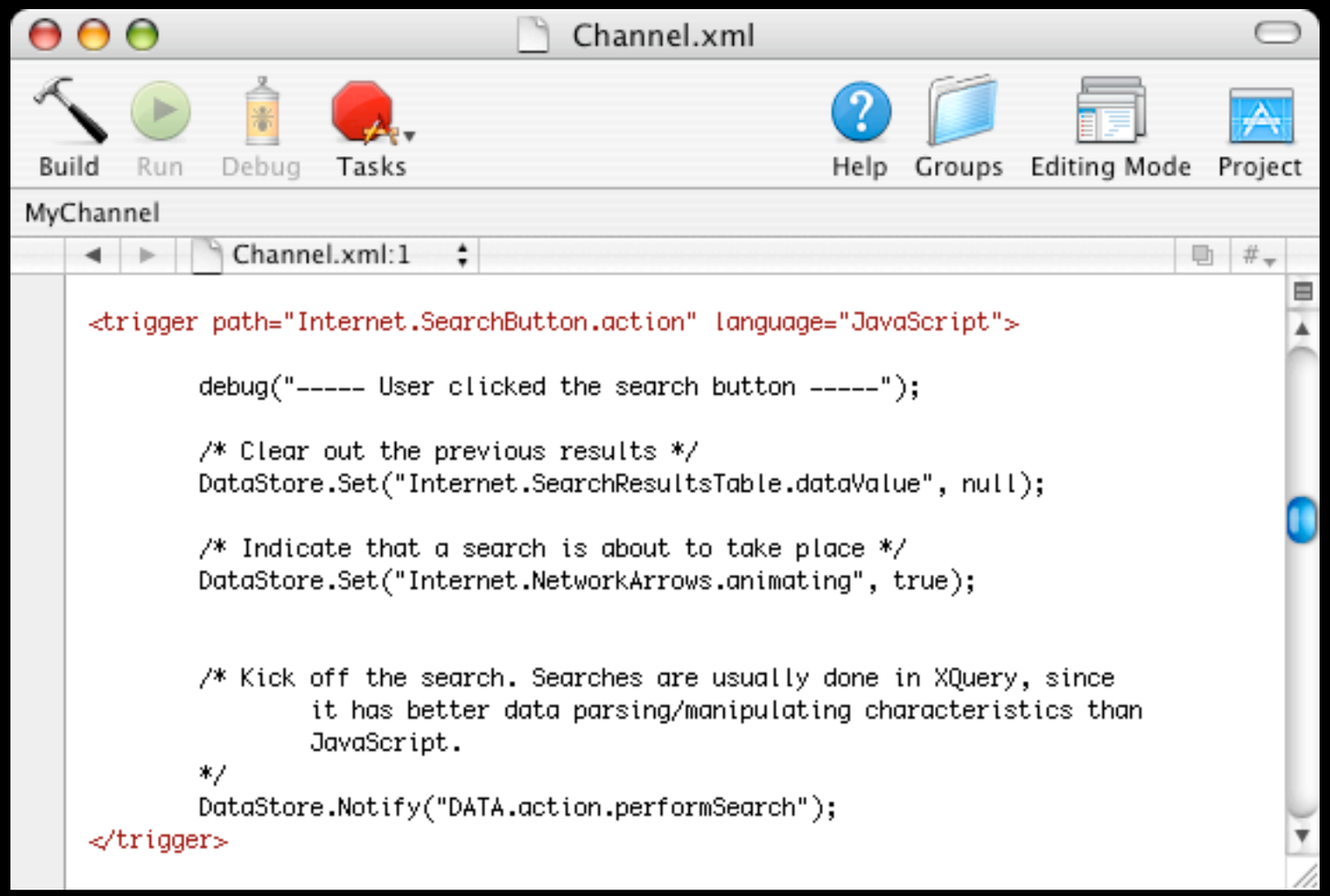
Enhance Code to Meet Your Needs

Triggers

- Triggers
  - Are executed when
    - value for data store path changes
    - data store path is notified
  - Use JavaScript

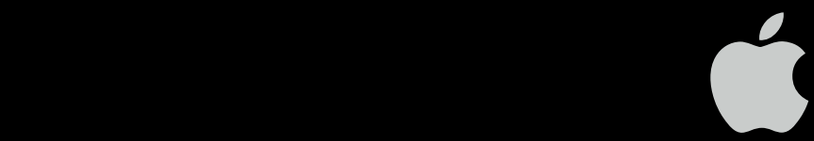


Triggers
Example



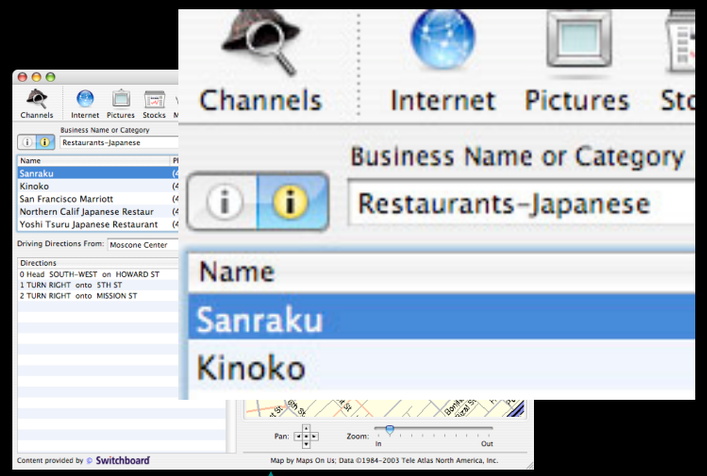
Triggers
Example

```
<trigger path="Internet.SearchButton.action"
   language="JavaScript">

   /* Clear out the previous results */
   DataStore.Set("Internet.SearchResultsTable.dataValue",
       null);

   /* Indicate that a search is about to take place */
   DataStore.Set("Internet.NetworkArrows.animating",
       true);

   /* Kick off the search */
   DataStore.Notify("DATA.action.performSearch");

</trigger>
```

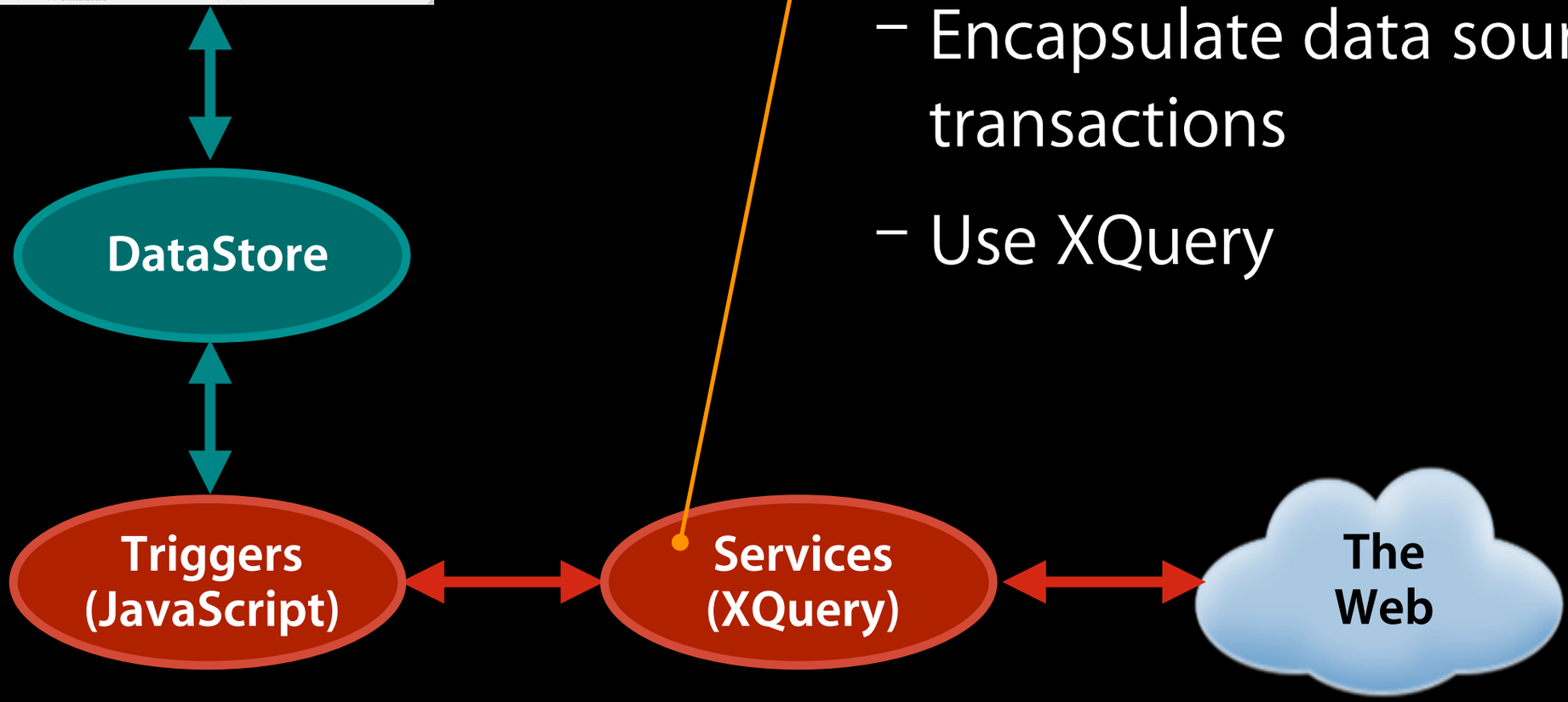


Enhance Code to Meet Your Needs

Services



- Services
  - Are called by triggers
  - Encapsulate data source (web) transactions
  - Use XQuery



Services
Example

soap.xml

Tasks | Help | Groups | Editing Mode

soap.xml:23

```
<!-- Copyright (c) 2002, Apple Computer, Inc. -->
<!-- All rights reserved.                      -->

<scripts>
<script language="XQuery">

{-- SOAPQuery send a SOAP query --}

define function SOAPQuery($query, $action, $soapAddress)
{
    {-- set the content type and the soap action for the request --}
    let $headers := dictionary(
        ("Content-Type", "text/xml; charset=utf-8"),
        ("SOAPAction", $action)
    )

    {-- send the request --}
    return http-post($soapAddress, $query, $headers)/DATA
}

</script>
</scripts>
```

Triggers
Example

```
define function SOAPQuery($query, $action, $soapAddress)
{
   {-- Set the content type and the soap action --}
   let $headers := dictionary(
      (“Content-Type”, “text/xml; charset=utf-8”),
      (“SOAPAction”, $action)
   )

   {-- Send the request --}
   return http-post($soapAddress, $query, $headers)/DATA
}
```

Developing Channels In Four Steps

1. Create
2. Customize
3. Debug
4. Deploy

Debug

- msg() and debug()
- XQuery and JavaScript Channels
- Enhanced Debug Menu

msg() and debug()

- Use msg() in XQuery code

```
let $doc := document(“http://www.apple.com/”)
let $log := msg($doc/source(.))
return $doc
```

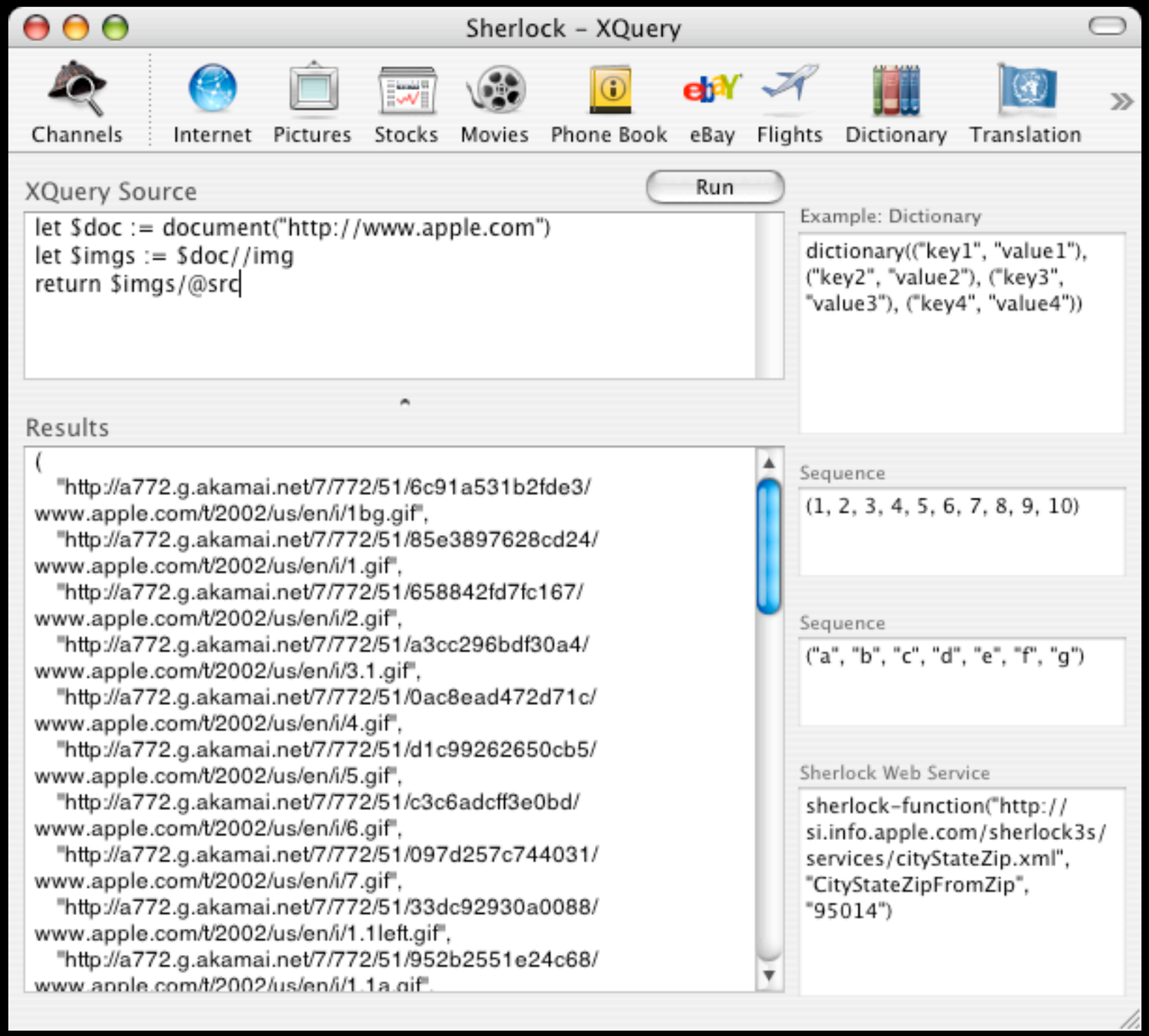
- Use debug() in JavaScript code

```
```

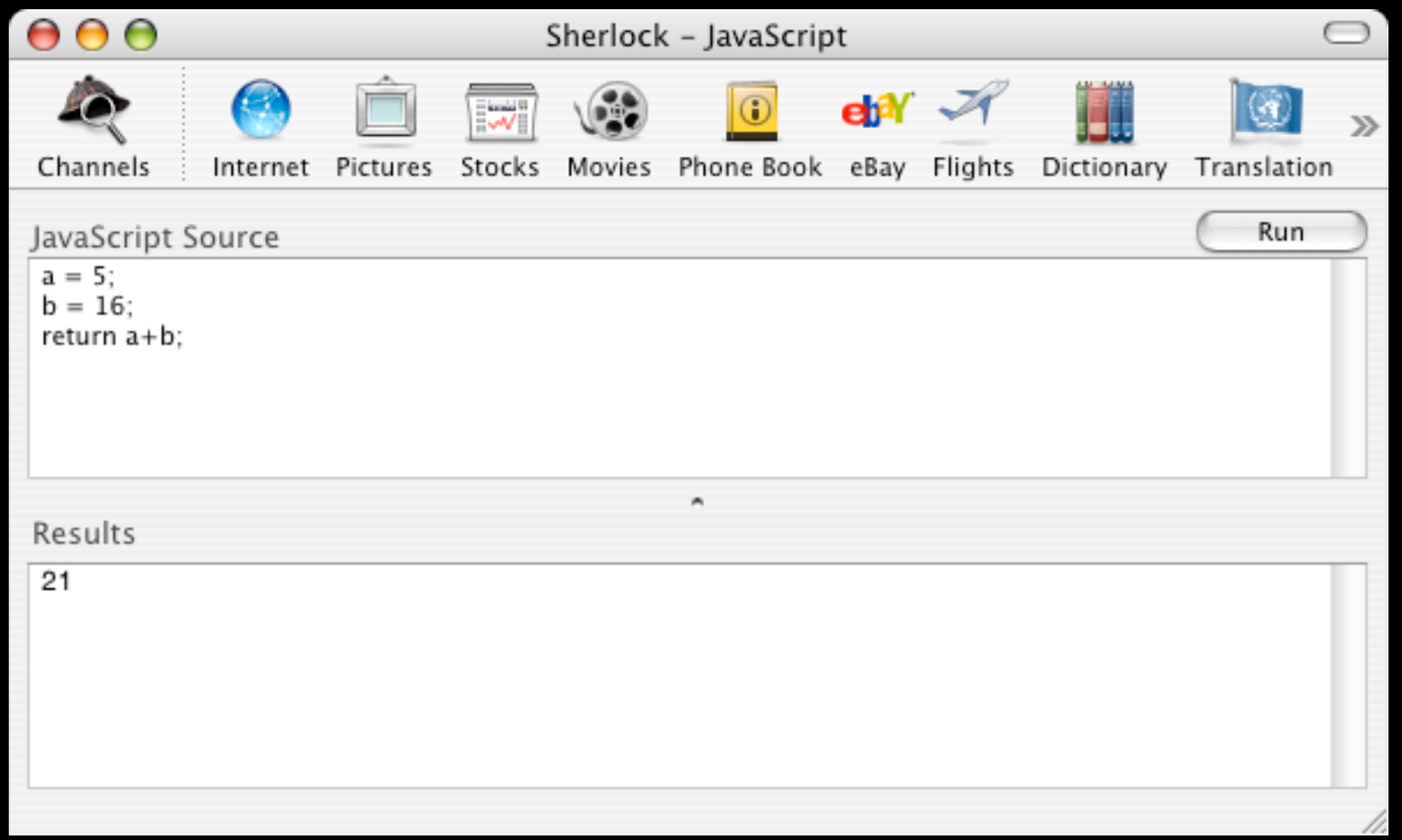
XQuery and JavaScript Channels

- Unit-test blocks of code in these channels

XQuery Channel

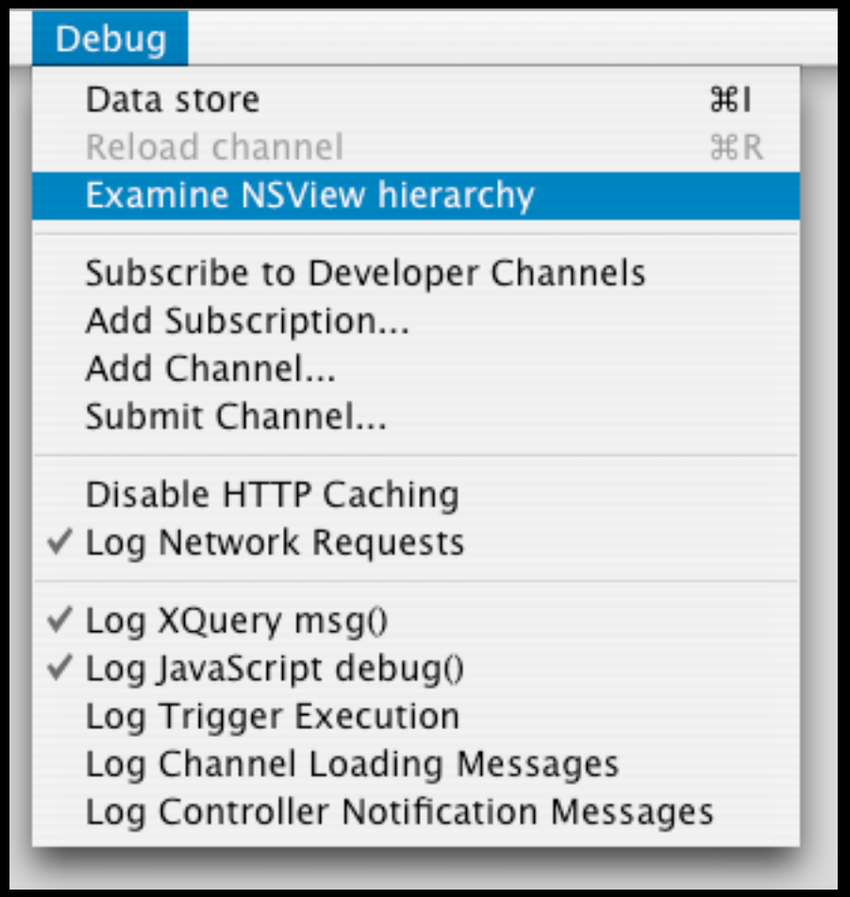


JavaScript Channel



Enhanced Debug Menu

Defaults write com.apple.Sherlock SherlockDebug 1



Developing Channels In Four Steps

1. Create
2. Customize
3. Debug
4. Deploy

Deploy

- Subscriptions and checkpoints
- FTP files to public web server
- Submit your channel to Apple!

Subscriptions and Checkpoints

- Subscription file lists channels you're deploying
  - Use when you ship multiple channels
  - Increases your channels' visibility
  - Convenient for the user
    - No need to add each channel one at a time
    - Control who sees your channels
    - Require specific Sherlock versions
    - Require specific geographic regions
- Checkpoints help Sherlock know what's changed

FTP Files to a Public Web Server

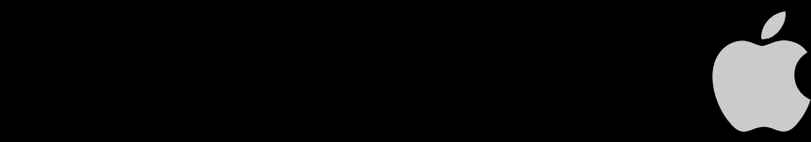
- Your company's, .Mac, or any other http server

Submit Your Channel to Apple

- Use “Submit Channel…” menu item
- Channel will appear in Other Channels collection

What's New for Sherlock Developers

- User Interface enhancements
  - NSTableView
  - NSComboBox
  - HTMLView
- Interapplication communication
- Address Book support
- Per-Channel Help
- Versioning



NSTableView

- Type selection in columns, or entire tables
- Control over row deletion
- Settable scroll position
- Checkboxes in table cells
- Sorting

NSComboBox

- Text selection on mouse down
- Greater control over list behavior

HTMLView

- Now using Web Kit
- Settable default charset
- Browsing within Sherlock

Interapplication Communication

- URLs allow channels to interact with other applications
  - addressbook:
  - ical:, webcal:
  - help:
- Send with JavaScript System.OpenURL()

Address Book Support

- SherlockAddressComboBox
- JavaScript AddressBook object
  - AddressBook.Add()
  - AddressBook.Search()

Per-Channel Help

- Provide online help to your users in Help Viewer
  - Supply an HTML help file
  - Sherlock does the rest!

Versioning

- Safe adoption of new Sherlock features
  - Check for existence of new JavaScript objects
  - Call channel-version()
  - Use version or maxVersion attributes

Advanced Topics

- Localized resources
- Printing
- Performance Tips

Localized Resources

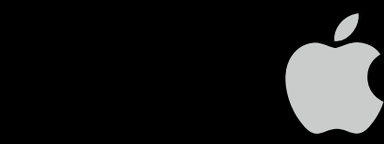
- localized_base_url
  - Tells Sherlock where to look for .lproj directories
- Accessing localized resources from your code
  - localized-resource()
  - localized-url()

Printing

- Use a custom print view
  - Condense complex channel views for 8.5” x 11” paper
- Be sure to set your channel’s mainViewIdentifier

Performance Tips

- Avoid <initialize> bloat
- Execute any given XPath only once
- FavorCacheUpdate for infrequently changed resources
- Stream returned data into view as it becomes available
- Show cached data first, then refresh



Wrap Up

Summary

- Deliver Web Services today using Sherlock Channels
  - Create quickly
  - Update on the web
  - Native Aqua UI
  - Link from Web or Dock
  - Built-In to Sherlock

Roadmap

403 Internet Technologies: Safari Overview
High-level overview of the technology used by Safari
The Presidio
Tue., 2:00pm

408 Apple Help Updates and How To's
Learn how to author Apple Help content
Mission
Tue., 5:00pm

411 Internet Technologies: Understanding Safari Technology and Web Standards
Details Safari and Web Kit's standards compliance
Nob Hill
Wed., 10:30am

731 Internet Application Development: The New Frontier
Learn how to develop apps for Macromedia Central
Marina
Fri., 2:00pm

Who to Contact

John Geleynse
Manager, Software Evangelism
geleynse@apple.com

http://developer.apple.com/wwdc2003/urls.html

For More Information

Reference Library

Documentation

Sherlock Developer Web Page
http://developer.apple.com/macosx/sherlock/

Sherlock Developer Mailing List
sherlock-channel-development@lists.apple.com
http://www.lists.apple.com/mailman/listinfo/sherlock-channel-development

Inside Mac OS X: Sherlock Channels
ADC Home > Documentation > Apple Applications > Documents

Sherlock Release Notes
ADC Home > Documentation > Release Notes

Reference Library

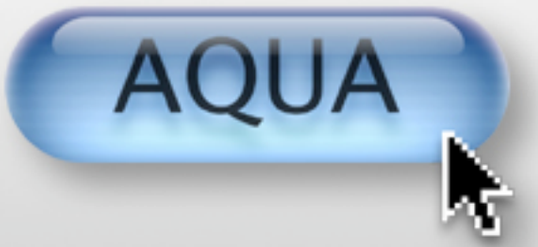
Documentation

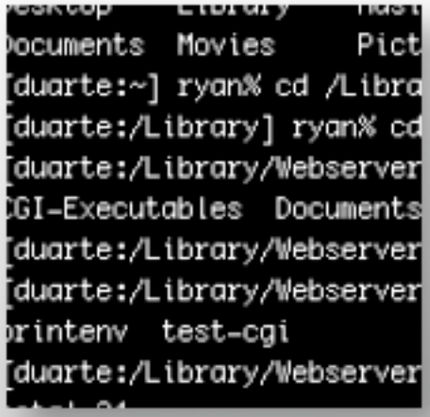
JavaScript Language Reference
http://www.ecma-international.org/publications/files/ecma-st/Ecma-262.pdf

XQuery Language Reference
http://www.w3.org/TR/2002/WD-xquery-20021115/
http://www.w3.org/TR/2002/WD-xquery-20020430/

XQuery Function Reference
http://www.w3.org/TR/2002/WD-xquery-operators-20021115/
http://www.w3.org/TR/2002/WD-xquery-operators-20020430/

Q&A





Wayne Loofbourrow
Jessica Kahn
Riley Howard, Eric Koebler, Sarah Wilkin

http://developer.apple.com/wwdc2003/urls.html

WWDC 2003

Worldwide Developers Conference